Document Title	AUTOSAR Diagnostic Extract Template
Document Owner	AUTOSAR
Document Responsibility	AUTOSAR
Document Identification No	673
Document Classification	Standard

Document Status	Final
Part of AUTOSAR Release	4.2.2

Document Change History

Release	Changed by	Description
4.2.2	AUTOSAR Release Management	• Minor corrections / clarifications / editorial changes; For details please refer to the ChangeDocumentation
4.2.1	AUTOSAR Release Management	• Initial Release

Disclaimer

Advice for users

AUTOSAR specifications may contain exemplary items (exemplary reference models, "use cases", and/or references to exemplary technical solutions, devices, processes or software).

Any such exemplary items are contained in the specifications for illustration purposes only, and they themselves are not part of the AUTOSAR Standard. Neither their presence in such specifications, nor any later documentation of AUTOSAR conformance of products actually implementing such exemplary items, imply that intellectual property rights covering such exemplary items are licensed under the same rules as applicable to the AUTOSAR Standard.

Table of Contents

Bibliography

[1] Specification of RTE Software
AUTOSAR_SWS_RTE

[2] Layered Software Architecture
AUTOSAR_EXP_LayeredSoftwareArchitecture

[3] Specification of ECU Configuration
AUTOSAR_TPS_ECUConfiguration

[4] ASAM MCD 2D ODX
http://www.asam.net
ASAM MCD-2D ODX v2.2.0.pdf

[5] Model Persistence Rules for XML
AUTOSAR_TR_XMLPersistenceRules

[6] System Template
AUTOSAR_TPS_SystemTemplate

[7] Specification of ECU Configuration Parameters (XML)
AUTOSAR_MOD_ECUConfigurationParameters

[8] Specification of Function Inhibition Manager
AUTOSAR_SWS_FunctionInhibitionManager

[9] Software Component Template
AUTOSAR_TPS_SoftwareComponentTemplate

[10] Basic Software Module Description Template
AUTOSAR_TPS_BSWModuleDescriptionTemplate

[11] Specification of Diagnostic Communication Manager
AUTOSAR_SWS_DiagnosticCommunicationManager

[12] Specification of Diagnostic Event Manager
AUTOSAR_SWS_DiagnosticEventManager

[13] Standardization Template
AUTOSAR_TPS_StandardizationTemplate

[14] Requirements on Diagnostic Extract Template
AUTOSAR_RS_DiagnosticExtractTemplate

[15] Unified diagnostic services (UDS) - Part 1: Specification and requirements (Release 2006-12)
http://www.iso.org

[16] Road vehicles – End-of-life activation of on-board pyrotechnic devices – Part 2: Communication requirements
http://www.iso.org

[17] SAE J1939-73 Application Layer - Diagnostics

[18] Software Process Engineering Meta-Model Specification
http://www.omg.org/spec/SPEM/2.0/

[19] Generic Structure Template
AUTOSAR_TPS_GenericStructureTemplate

1 Introduction

1.1 Overview

The distributed nature of an AUTOSAR ECU development requires an optimized capturing of information. In particular, diagnostic information (i.e. DEM and DCM configuration) shall be captured only once by the person with the best knowledge and therefore being able to take responsibility better than one centralized individual.

In the configuration approach before the advent of the `DiagnosticExtract`, the Basic Software Modules DCM and DEM are entirely configured centrally. During integration, all SW-Cs above the RTE [1] (Application Software) introduce ports to be connected to the BSW modules [2]. Additionally, SW-Cs express needs which shall be fulfilled by the BSW.

The market shows a high demand for transferring diagnostic demands of the OEM-specific configuration process to their tier-1 suppliers.

In the past, due to the absence of integral options, many different file formats like ODX or ECU-C [3] are often used. But neither ODX nor ECU-C is well suited to transfer this information.

For example, ODX [4] lacks in fault memory details and ECU-C (which was never designed for becoming the vehicle for data exchange between different organizations) has a very generic nature that renders the enforcement of a strict model formalization very difficult.

On top of that, the integration of ECU-C definitions into an existing configuration (especially the PDUs) cannot be fully automated.

Therefore, the obvious solution approach has been to define a new standardized AUTOSAR exchange format on diagnostic functionality that can be used similar to a System Description, formalized as an ARXML [5] file.

In this spirit, the configuration of diagnostic functionality becomes similar to the configuration of the communication part within the System Description [6].

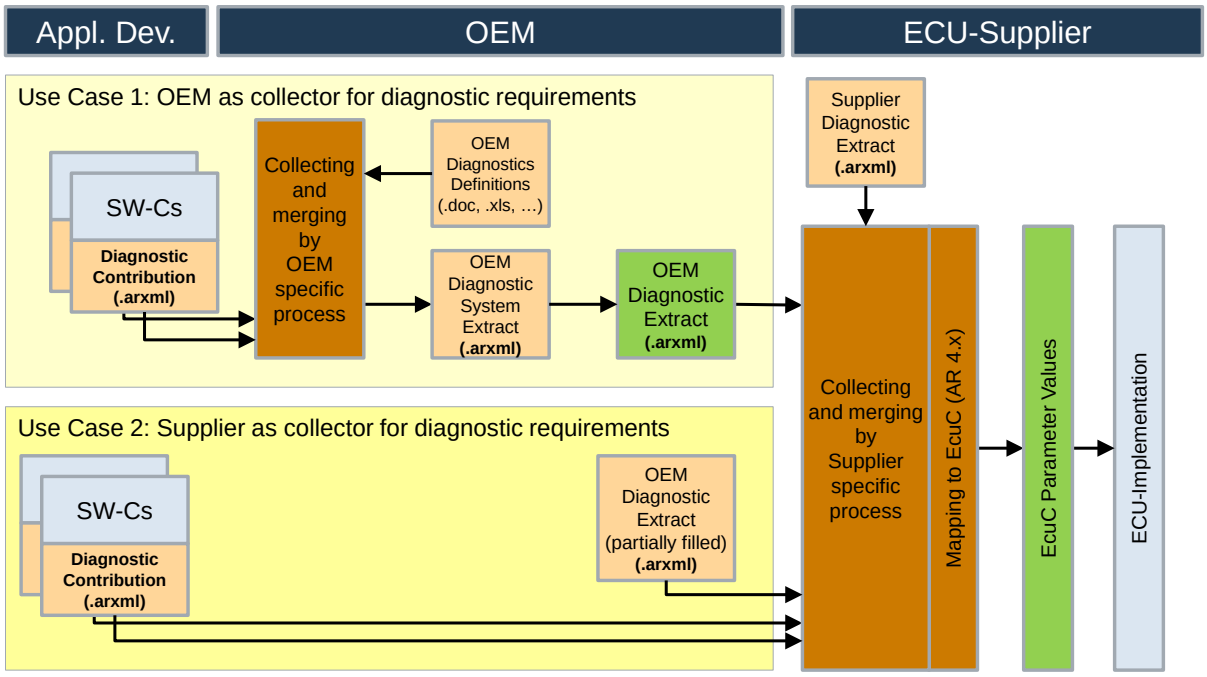

Figure 1.1: Scope of this document in the ECU Development work-flow

Figure 1.1 shows the future configuration process of diagnostics for two generalized use cases. This process involves three parties:

- **OEM** or diagnostic requester
- **Application Developer** or Application Developer
- **ECU-Supplier** or integrator

1.1.1 OEM

The **OEM** or requester of diagnostic data uses the `DiagnosticExtract` to define the diagnostic interfaces of one or multiple ECUs. It may also define some `InternalBehavior`s as requirements for the **ECU-Supplier** or **Application Developer**

- Defines the values of the DTCs
- Defines the UDS services and sub-services supported by the ECUs
- Defines the required events needed by a specific composition implemented by an **Application Developer**

NOTE: Only as example, this document does not define a specific ownership of each element.

In the first use case, the `DiagnosticExtract` is used to exchange information which is transformed into the ECU-C configuration (M2 to M1 mapping, see also [3] and [7]).

Second, the **OEM** uses the DiagnosticExtract to document requirements to be implemented by a supplier. These requirements are expressed in textual language and can not be mapped directly to any ECU-C configuration parameters (no M2 to M1 mapping possible).

1.1.2 Application Developer

The Application Developers implement their **SWCs** with the corresponding **SWC Description**. With the introduction of this concept, the Application Developer has the possibility to provide diagnostic information relevant to the SWCs as part of the DiagnosticExtract.

The Application Developer may also receive some input as requirement from the **OEM** within the DiagnosticExtract in textual form as for example:

- Definition of the content of a specific ReadDataByIdentifier implemented by this SWC
- Definition of the events needed for this SWC

NOTE: Only as example, this document does not define a specific ownership of each element.

In the first use case, the Application Developer defines the parameters of a specific ReadDataByIdentifier, i.e. the content of the diagnostic request but not the DID itself. The DID of this command will usually be defined by the **OEM**.

Secondly, the SWC events including information like Debouncing and OperationCycle may be defined by the **Application Developer**. The **Application Developer**may also define events and diagnostic jobs which are not needed by a specific **OEM** but for another one.

Suppliers may use the same software for multiple **OEMs** and need to reuse it. This implies that some DiagnosticExtract information coming from a SWC may be ignored during the integration if not needed for a specific project.

1.1.3 ECU-Supplier

The **ECU-Supplier** or integrator receives one or several DiagnosticExtract files from the **OEM** and from multiple **Application Developers**. The main goal of the **Integrator** is to integrate all delivered DiagnosticExtract and to generate the ECU-C configuration from it.

Since this concept does not define a specific ownership for each element like DIDs, parameters of a UDS service, Events, Sessions, etc. the integrator has to ensure that the complete information is still valid after merging it.

- Mapping of DTCs to Events

- Merge of Events
- Mapping of services

Some DTCs may already be mapped to events - especially in cases where both come from the same party. But if the DTCs are defined by the **OEM**and the SW implemented by other supplier acting as an **Application Developer** the integrator has to ensure that both are mapped together.

In some cases, an Event may be defined multiple times. An **OEM** defines the Events which shall be implemented by an **Application Developer**. A Supplier implements a SWC which will be used in multiple projects and which also detects this type of error and also defines this same event.

Both events may have different naming but the same meaning. The integrator has to detect this redundancy during the integration and merge them together.

In another case, the **OEM** requires a specific `ReadDataByIdentifier` and an **Application Developer** implements it. If the implementation is performed for one specific project only, the **Application Developer** may map the DID from the **OEM** to the already defined job in their SWC.

In other cases in which the **Application Developer** implements a generic diagnostic job, it will be a task of the **ECU-Supplier** to merge this information and to map the jobs to the corresponding DID.

1.1.4 Exchanging of Files

During an ECU development project, the three main roles (OEM, Application Developer, ECU Integrator) exchange `DiagnosticExtract` files. The timing and frequency of exchanges and the content in each of these exchanged files is highly dependent on the individual project setup and situation.

Therefore, the `DiagnosticExtract` format has been designed to allow for gradual enrichment of definitions contributed at largely arbitrary points in time by the different roles in order to meet the needs of "Decentralized Configuration".

For any exchange path between any two roles, the same file format based on the `DiagnosticExtract` template is used. It is then up to a company specific process and tooling to merge the collected `DiagnosticExtract` files while resolving conflicts (contradictions, redundancies etc.).

As final result, a consistent and complete `DiagnosticExtract` file is available which is input to derivation of the configuration for the diagnostic modules of the Basic Software.

1.1.5 Relationship to SWC Service Needs

Before the introduction of the Diagnostic Extract, Service Needs were used to describe diagnostic requirements on SWC level. These configuration requirements are referenced to the related BSW module DCM or DEM in order to provide the corresponding configuration on BSW level.

The usage of Service Needs is only possible on Atomic SWC level whereas the assignment of diagnostic demands must be possible on Composition level.

The `SWC Service Needs` within the `SWC Description` are still to be used along with the `DiagnosticExtract` in order to annotate the SWC ports which are relevant for further mapping and handling as defined by the `DiagnosticExtract`.

From SWC Developer's perspective, the `DiagnosticExtract` is therefore used partially as substitution and partially as extension of the `SWC Service Needs`. The reasoning for "substitution" is the avoidance of redundant diagnostic definitions.

Since some diagnostic properties potentially definable by SWC Developers are not covered in `SWC Service Needs`, the `DiagnosticExtract` can also be viewed as "extension" to the original purpose of `SWC Service Needs`.

1.1.6 Recommendation and Hints

Multiple parties may have different understanding of which parts shall be provided by each one. There is no defined rule to indicate who is responsible for each part. At the end, it is the **ECU-Supplier** in his role as integrator who has to ensure that all mappings are done and that the ECU runs as expected by the **OEM**.

In case the **OEM** does not have his own diagnostic requirements, the **ECU-Supplier** has to provide the complete `DiagnosticExtract`. In this case, the **OEM** may only receive the `DiagnosticExtract` as part of the delivery. The process itself how the parties work with this format is not defined within this specification.

Figure 1.1 shows a recommended way how to handle the `DiagnosticExtract` between the different parties. In use case 1, some SWCs are implemented by the **OEM** (or by a Supplier of the OEM) and the first merging of `DiagnosticExtract` data occurs at the **OEM**.

In use case 2, the **OEM** provides the diagnostic requirements via `DiagnosticExtract` and multiple **Application Developer**provide information related to their implementation, the merging is performed completely by the **ECU-Supplier**.

Also a combination of use cases 1 and 2 is allowed. Also the **ECU-Supplier** may implement some part of the SW inclusive their corresponding `DiagnosticExtract`.

1.1.7 Limitations

This first release of the DiagnosticExtract template focuses on defining diagnostic requirements a single ECU only. That means that currently, distributed diagnostic functionality for a system or partial system consisting of multiple ECUs cannot be defined using the DiagnosticExtract template.

In future releases, the DiagnosticExtract template is expected to be extended to also cover configuration of distributed diagnostic functionality. Similar to the description of communication dependencies in the System Description, it shall be possible to describe diagnostic demands on system level to derive the diagnostic demands for a specific ECU from this description.

Additionally, the configuration of **Fim** [8] using DiagnosticExtract has been postponed to a future AUTOSAR release. It is expected that the configuration of the **Fim** Module will be treated in a similar way as the **Dem** and **Dcm** configuration using the DiagnosticExtract.

Further, the DiagnosticExtract does not support OBD-related configuration parts of DCM and DEM including WWH-OBD (World-Wide Harmonized OBD).

The configuration of mode rules is currently not supported by the DiagnosticExtract.

The DiagnosticExtract does currently not support the configuration of multiple diagnostic protocols (UDS, OBD, J1939) and the corresponding protocol prioritization. This means, that only UDS according to ISO14229-1 is supported and another protocol that is implemented in parallel needs to be configured manually.

In general, the DiagnosticExtract does not support process-related parts to document the maturity of diagnostic configuration data. This means, that a data object cannot be marked as "draft" or "released". This issue needs to be solved by AUTOSAR within a general concept in a future release. Therefore, it does not make sense to introduce a solution for diagnostics only.

1.2 Scope

This document describes the formal description of contributions to the diagnostic configuration.

On the level of meta-modeling, the content described in this document conceptually relates to the definition of SwcServiceDependency resp. BswServiceDependency, as defined by the Software Component Template [9] resp. Basic Software ModuleDescription Template [10].

Further relations exist to the specification of communication in AUTOSAR systems as described by the System Template [6].

Further relations exist to the specification of the Diagnostic Communication Manager [11] as well as to the Diagnostic Event Manager [12].

The relation of the DiagnosticExtract to the rest of the AUTOSAR meta-model is sketched in Figure 1.2.

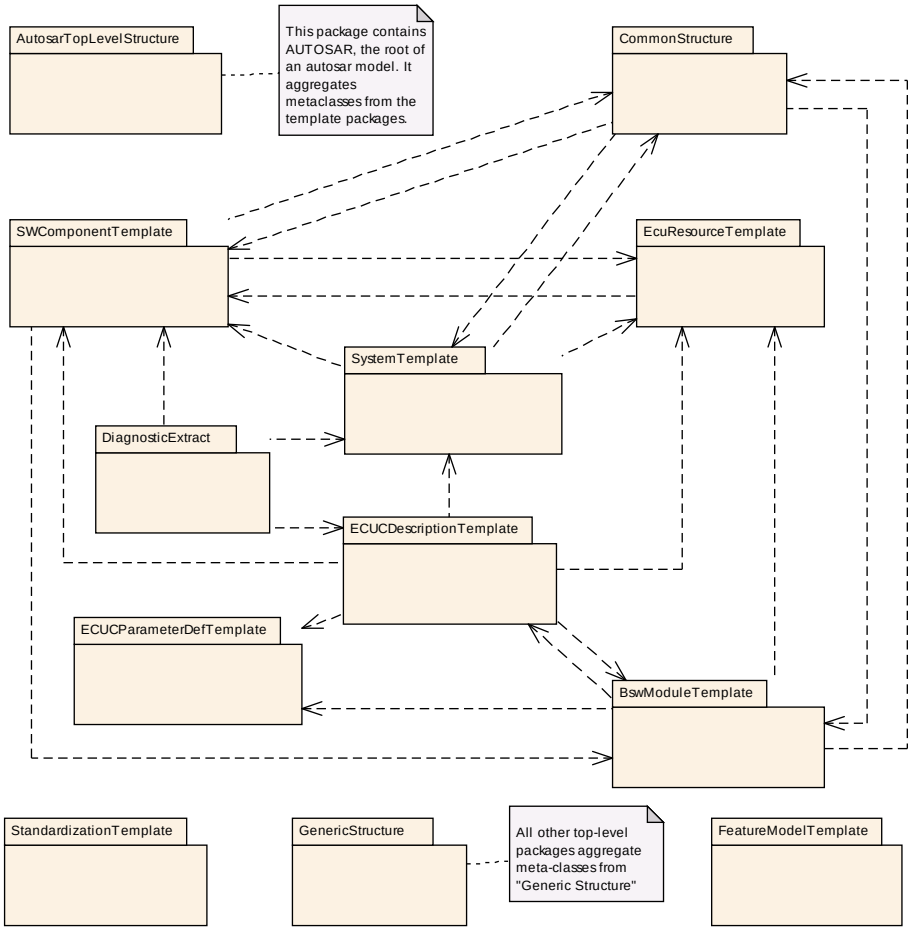

Figure 1.2: The relation of the DiagnosticExtract to the rest of the AUTOSAR meta-model

1.3 Abbreviations

The following table contains a list of abbreviations used in the scope of this document along with the spelled-out meaning of each of the abbreviations.

Abbreviation	*meaning*
API	Application Programming Interface
BSW	Basic Software
BswM	Basic Software Manager
CAN	Controller Area Network
CSE	Codes for Scaling Units
DEM	Diagnostics Communication Manager
DCY	Driving Cycle

DEM	Diagnostics Event Manager
DID	Diagnostic Identifier
DTC	Diagnostic Trouble Code
DTR	Diagnostic Test Result
DoIP	Diagnostics over IP
ECU	Electrical Control Unit
ECUC	ECU Configuration
FID	Function Identifier
FIM	Function Inhibition Manager
GID	Group Identifier
ID	Identifier
IO	Input/Output
IP	Internet Protocol
IUMPR	In-Use Monitor Performance Ratio
ISO	International Organization for Standardization
LIN	Local Interconnect Network
OBD	On-Board Diagnostic
ODX	Open Diagnostic Data Exchange
OEM	Original Equipment Manufacturer
PDU	Protocol Data Unit
PID	Parameter Identifier
PTO	Power Take Off
RA	Routing Activation
RAM	Random Access Memory
RID	Routine Identifier
ROE	Response on Event
ROM	Read-Only Memory
RTE	Run-TIme Environment
RS	Requirements Specification
RX	Receive
SW	Software
SWC	Software Component
SWCD	Software Component Description
TID	Test Identifier
TPS	Template Specification
TX	Transmit
SWS	Software Specification
UDS	Unified Diagnostic Services
UML	Unified Modeling Language
VFB	Virtual Functional Bus
VIN	Vehicle Identification Number
WWH-OBD	World-Wide Harmonized On-Board Diagnostics
XML	Extensible Markup Language
XSD	XML Schema Definition

Table 1.1: Abbreviations used in the scope of this Document

1.4 Document Conventions

Technical terms are typeset in mono spaced font, e.g. `PortPrototype`. As a general rule, plural forms of technical terms are created by adding "s" to the singular form, e.g.

PortPrototypes. By this means the document resembles terminology used in the AUTOSAR XML Schema.

This document contains constraints in textual form that are distinguished from the rest of the text by a unique numerical constraint ID, a headline, and the actual constraint text starting after the ⌈ character and terminated by the ⌋ character.

The purpose of these constraints is to literally constrain the interpretation of the AUTOSAR meta-model such that it is possible to detect violations of the standardized behavior implemented in an instance of the meta-model (i.e. on M1 level).

Makers of AUTOSAR tools are encouraged to add the numerical ID of a constraint that corresponds to an M1 modeling issue as part of the diagnostic message issued by the tool.

The attributes of the classes introduced in this document are listed in form of class tables. They have the form shown in the example of the top-level element AUTOSAR:

Class	**AUTOSAR**			
Package	M2::AUTOSARTemplates::AutosarTopLevelStructure			
Note	Root element of an AUTOSAR description, also the root element in corresponding XML documents. **Tags:** xml.globalElement=true			
Base	ARObject			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
adminData	AdminData	0..1	aggr	This represents the administrative data of an Autosar file. **Tags:** xml.sequenceOffset=10
arPackage	ARPackage	*	aggr	This is the top level package in an AUTOSAR model. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=shortName, variation Point.shortLabel vh.latestBindingTime=blueprintDerivationTime xml.sequenceOffset=30
introductio n	Documentation Block	0..1	aggr	This represents an introduction on the Autosar file. It is intended for example to rpresent disclaimers and legal notes. **Tags:** xml.sequenceOffset=20

Table 1.2: AUTOSAR

The first rows in the table have the following meaning:

Class: The name of the class as defined in the UML model.

Package: The UML package the class is defined in. This is only listed to help locating the class in the overall meta model.

Note: The comment the modeler gave for the class (class note). Stereotypes and UML tags of the class are also denoted here.

Base Classes: If applicable, the list of direct base classes.

The headers in the table have the following meaning:

Attribute: The name of an attribute of the class. Note that AUTOSAR does not distinguish between class attributes and owned association ends.

Datatype: The datatype of an attribute of the class.

Mul.: The assigned multiplicity of the attribute, i.e. how many instances of the given data type are associated with the attribute.

Kind: Specifies, whether the attributes is aggregated in the class (`aggr`), an UML attribute in the class (`attr`), or just referenced by it (`ref`). Instance references are also indicated (`iref`) in this field.

Note: The comment the modeler gave for the class attribute (role note). Stereotypes and UML tags of the class are also denoted here.

The verbal forms for the expression of obligation specified in [TPS_STDT_00053] shall be used to indicate requirements, see Standardization Template, chapter Support for Traceability ([13]).

The representation of requirements in AUTOSAR documents follows the table specified in [TPS_STDT_00078], see Standardization Template, chapter Support for Traceability ([13]).

1.5 Requirements Tracing

The following table references the requirements specified in [14] and links to the fulfillment of these.

Requirement	Description	Satisfied by
[RS_DEXT_00001]	Diagnostic data exchange	[TPS_DEXT_01003] [TPS_DEXT_01004] [TPS_DEXT_01005] [TPS_DEXT_01007] [TPS_DEXT_01008] [TPS_DEXT_01014] [TPS_DEXT_01019] [TPS_DEXT_01020] [TPS_DEXT_01022] [TPS_DEXT_01023] [TPS_DEXT_01024] [TPS_DEXT_01025] [TPS_DEXT_01026] [TPS_DEXT_01027] [TPS_DEXT_01028] [TPS_DEXT_01029] [TPS_DEXT_01046] [TPS_DEXT_01055] [TPS_DEXT_01056] [TPS_DEXT_01057] [TPS_DEXT_01058] [TPS_DEXT_01059] [TPS_DEXT_01060] [TPS_DEXT_01062] [TPS_DEXT_01063] [TPS_DEXT_01066] [TPS_DEXT_01069] [TPS_DEXT_01075]

[RS_DEXT_00002]	Distributed software development process	[TPS_DEXT_01003] [TPS_DEXT_01004] [TPS_DEXT_01005] [TPS_DEXT_01038] [TPS_DEXT_01047] [TPS_DEXT_01055] [TPS_DEXT_01062] [TPS_DEXT_01063]
[RS_DEXT_00003]	SessionControl	[TPS_DEXT_01039] [TPS_DEXT_01045] [TPS_DEXT_01081] [TPS_DEXT_01082]
[RS_DEXT_00004]	ECUReset	[TPS_DEXT_01019] [TPS_DEXT_01020] [TPS_DEXT_01021] [TPS_DEXT_01045] [TPS_DEXT_01056]
[RS_DEXT_00005]	ClearDiagnosticInformation	[TPS_DEXT_01022] [TPS_DEXT_01045]
[RS_DEXT_00006]	ReadDTCInformation	[TPS_DEXT_01034] [TPS_DEXT_01045] [TPS_DEXT_01060]
[RS_DEXT_00007]	ReadDataByIdentifier	[TPS_DEXT_01045] [TPS_DEXT_01050] [TPS_DEXT_01054]
[RS_DEXT_00008]	ReadMemoryByAddress	[TPS_DEXT_01024] [TPS_DEXT_01045]
[RS_DEXT_00009]	SecurityAccess	[TPS_DEXT_01036] [TPS_DEXT_01037] [TPS_DEXT_01038] [TPS_DEXT_01045] [TPS_DEXT_01053]
[RS_DEXT_00010]	CommunicationControl	[TPS_DEXT_01029] [TPS_DEXT_01030] [TPS_DEXT_01031] [TPS_DEXT_01032] [TPS_DEXT_01045] [TPS_DEXT_01057] [TPS_DEXT_01074]
[RS_DEXT_00011]	ReadDataByPeriodicIdentifier	[TPS_DEXT_01045] [TPS_DEXT_01059]
[RS_DEXT_00012]	DynamicallyDefineDataIdentifier	[TPS_DEXT_01045] [TPS_DEXT_01058]
[RS_DEXT_00013]	WriteDataByIdentifier	[TPS_DEXT_01045] [TPS_DEXT_01050] [TPS_DEXT_01054]
[RS_DEXT_00014]	IOControl	[TPS_DEXT_01015] [TPS_DEXT_01016] [TPS_DEXT_01017] [TPS_DEXT_01018] [TPS_DEXT_01045] [TPS_DEXT_01051]
[RS_DEXT_00015]	RoutineControl	[TPS_DEXT_01035] [TPS_DEXT_01045] [TPS_DEXT_01049] [TPS_DEXT_01077] [TPS_DEXT_01078] [TPS_DEXT_01079] [TPS_DEXT_01080]
[RS_DEXT_00016]	RequestDownload	[TPS_DEXT_01027] [TPS_DEXT_01045]
[RS_DEXT_00017]	RequestUpload	[TPS_DEXT_01028] [TPS_DEXT_01045]
[RS_DEXT_00018]	TransferData	[TPS_DEXT_01026] [TPS_DEXT_01045]
[RS_DEXT_00019]	RequestTransferExit	[TPS_DEXT_01025] [TPS_DEXT_01045]
[RS_DEXT_00020]	WriteMemoryByAddress	[TPS_DEXT_01023] [TPS_DEXT_01045]
[RS_DEXT_00021]	ControlDTCSetting	[TPS_DEXT_01045] [TPS_DEXT_01075] [TPS_DEXT_01076]
[RS_DEXT_00022]	ResponseOnEvent	[TPS_DEXT_01033] [TPS_DEXT_01045]
[RS_DEXT_00023]	Configuration of events	[TPS_DEXT_01048] [TPS_DEXT_01067] [TPS_DEXT_01068] [TPS_DEXT_01069] [TPS_DEXT_01083] [TPS_DEXT_01084] [TPS_DEXT_01085] [TPS_DEXT_03002] [TPS_DEXT_03003] [TPS_DEXT_03004] [TPS_DEXT_03005] [TPS_DEXT_03007] [TPS_DEXT_03011] [TPS_DEXT_03015] [TPS_DEXT_03016]
[RS_DEXT_00024]	Configuration of DTCs	[TPS_DEXT_01064] [TPS_DEXT_01065] [TPS_DEXT_01066] [TPS_DEXT_01086] [TPS_DEXT_03000] [TPS_DEXT_03003] [TPS_DEXT_03012] [TPS_DEXT_03013] [TPS_DEXT_03014]

[RS_DEXT_00026]	Enable Conditions	[TPS_DEXT_03015] [TPS_DEXT_03018]
[RS_DEXT_00027]	Storage Conditions	[TPS_DEXT_03001] [TPS_DEXT_03006] [TPS_DEXT_03010] [TPS_DEXT_03016] [TPS_DEXT_03019]
[RS_DEXT_00028]	Enable Condition Groups	[TPS_DEXT_01084] [TPS_DEXT_03010] [TPS_DEXT_03015]
[RS_DEXT_00029]	Storage Condition Groups	[TPS_DEXT_01084] [TPS_DEXT_03016]
[RS_DEXT_00030]	Assignment of Enable Condition Groups	[TPS_DEXT_03010]
[RS_DEXT_00031]	Assignment of Storage Condition Group	[TPS_DEXT_03010]
[RS_DEXT_00032]	Configuration of Extended Data Records	[TPS_DEXT_03008]
[RS_DEXT_00033]	Configuration of Snapshot Records	[TPS_DEXT_03009]
[RS_DEXT_00034]	Description of Data Identifiers	[TPS_DEXT_01000] [TPS_DEXT_01001] [TPS_DEXT_01002] [TPS_DEXT_01017] [TPS_DEXT_01050] [TPS_DEXT_01054] [TPS_DEXT_01072]
[RS_DEXT_00035]	Description of Dynamic Data Identifiers	[TPS_DEXT_01000]
[RS_DEXT_00036]	Description of Routine Identifiers	[TPS_DEXT_01088]
[RS_DEXT_00037]	Description of I/O Identifiers	[TPS_DEXT_01089]
[RS_DEXT_00038]	Description of array data types	[TPS_DEXT_01001] [TPS_DEXT_01002]
[RS_DEXT_00039]	Diagnostic Service Table	[TPS_DEXT_01006]
[RS_DEXT_00040]	Diagnostic Sessions	[TPS_DEXT_01011] [TPS_DEXT_01081] [TPS_DEXT_01082]
[RS_DEXT_00041]	Access Permissions	[TPS_DEXT_01012] [TPS_DEXT_01052] [TPS_DEXT_01061] [TPS_DEXT_01062] [TPS_DEXT_01063] [TPS_DEXT_01071]
[RS_DEXT_00042]	Security Levels	[TPS_DEXT_01012] [TPS_DEXT_01038] [TPS_DEXT_01053]
[RS_DEXT_00043]	Description of data elements	[TPS_DEXT_03020]
[RS_DEXT_00045]	Textual descriptions	[TPS_DEXT_01064] [TPS_DEXT_01065] [TPS_DEXT_01066] [TPS_DEXT_01067] [TPS_DEXT_01068] [TPS_DEXT_01069] [TPS_DEXT_01071]
[RS_DEXT_00047]	Custom Diagnostic Service	[TPS_DEXT_01009] [TPS_DEXT_01010] [TPS_DEXT_01021] [TPS_DEXT_01030] [TPS_DEXT_01031]
[RS_DEXT_00048]	Diagnostic Properties that are specific for one ECU	[TPS_DEXT_01073]
[RS_DEXT_00049]	Properties of individual diagnostic services	[TPS_DEXT_01013] [TPS_DEXT_01052] [TPS_DEXT_01061]
[RS_DEXT_00050]	Properties of all diagnostic services of a given kind	[TPS_DEXT_01061]

[RS_DEXT_00051]	Subfunctions of Diagnostic Services	[TPS_DEXT_01013] [TPS_DEXT_01014] [TPS_DEXT_01018] [TPS_DEXT_01019] [TPS_DEXT_01020] [TPS_DEXT_01021] [TPS_DEXT_01022] [TPS_DEXT_01023] [TPS_DEXT_01024] [TPS_DEXT_01025] [TPS_DEXT_01026] [TPS_DEXT_01027] [TPS_DEXT_01028] [TPS_DEXT_01029] [TPS_DEXT_01030] [TPS_DEXT_01031] [TPS_DEXT_01034] [TPS_DEXT_01039] [TPS_DEXT_01056] [TPS_DEXT_01057] [TPS_DEXT_01058] [TPS_DEXT_01059] [TPS_DEXT_01060] [TPS_DEXT_01075] [TPS_DEXT_01076] [TPS_DEXT_01078]
[RS_DEXT_00052]	Mapping of diagnostic services to the PortPrototypes of ApplicationSwComponentTypes	[TPS_DEXT_01040] [TPS_DEXT_01041] [TPS_DEXT_01042] [TPS_DEXT_01043] [TPS_DEXT_01044] [TPS_DEXT_01049] [TPS_DEXT_01050] [TPS_DEXT_01051] [TPS_DEXT_03002] [TPS_DEXT_03007] [TPS_DEXT_03017] [TPS_DEXT_03018] [TPS_DEXT_03019] [TPS_DEXT_03020]
[RS_DEXT_00053]	Debouncing of diagnostic events	[TPS_DEXT_01048] [TPS_DEXT_03004] [TPS_DEXT_03005] [TPS_DEXT_03017]
[RS_DEXT_00054]	Operation cycles	[TPS_DEXT_01086] [TPS_DEXT_01087]
[RS_DEXT_00055]	Aging	[TPS_DEXT_03021]
[RS_DEXT_00056]	Indicator	[TPS_DEXT_03022]
[RS_DEXT_00057]	RequestFileTransfer	[TPS_DEXT_01090]

Table 1.3: RequirementsTracing

2 Use Cases

2.1 Use cases for diagnostic data exchange

The basic usage of the `DiagnosticExtract` is the exchange of diagnostic data between the different parties involved in the diagnostic development process to allow the configuration of the DCM and the DEM and to provide the description of corresponding application interfaces to implement diagnostic services and fault handling.

2.2 Configuration of DCM

The configuration of the DCM includes the setup of diagnostic services and the assignment of data objects provided by one or more software components (e.g. Composition 1, Composition 2).

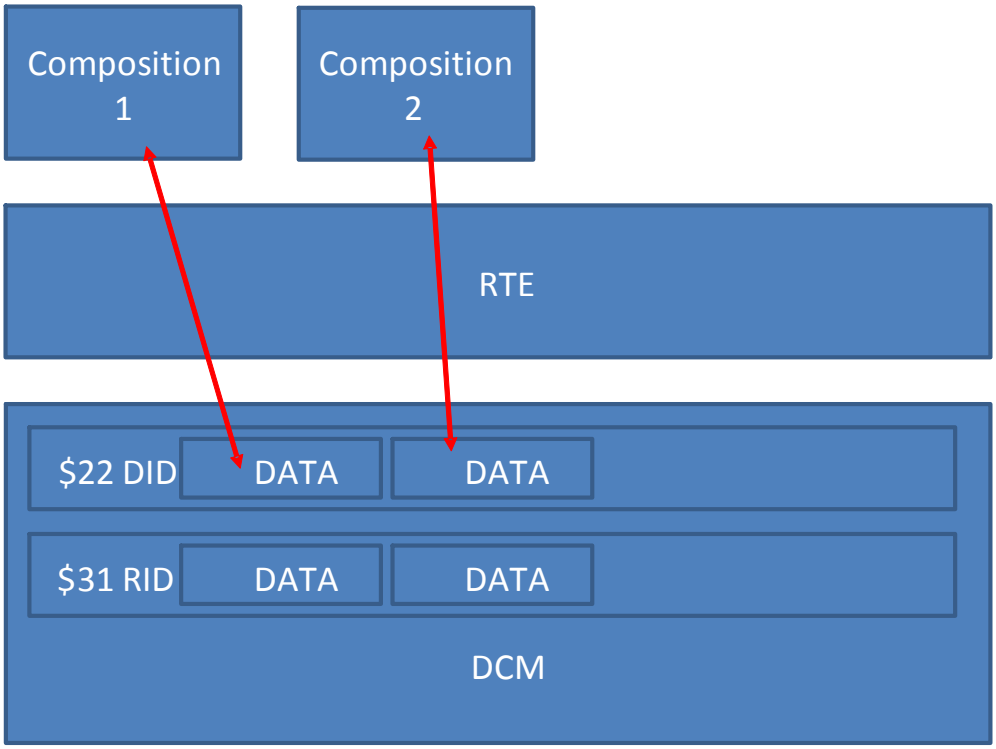

Figure 2.1: Assignment of DCM-related data objects

2.3 Configuration of DEM

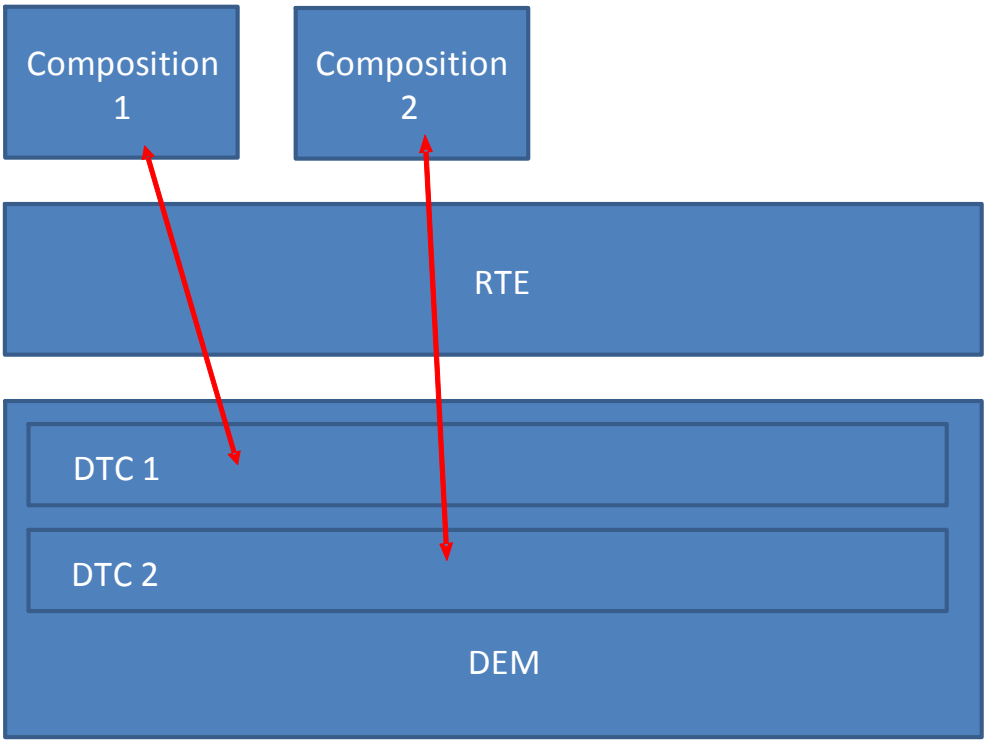

Figure 2.2: Assignment of DEM-related data objects

The configuration of the DEM includes fault memory data (DTCs and environmental data) and the assignment of corresponding data by one or more SwComponentTypes (e.g. Composition 1, Composition 2).

As already explained, the diagnostic development process is distributed among different parties. On the one hand side, the OEM needs to describe the general requirements for a diagnostic system that have to be implemented by an ECU:

1. Depending on the diagnostic system, the OEM can provide a completely or partly-filled DiagnosticExtract including the description of PortInterfaces:
   - Integrator/SWC developer (OEM or Tier 1) is responsible for the completion (detailing of predefined diagnostic content).
   - Integrator/SWC developer (OEM or Tier 1) is responsible for the specific configuration of diagnostic content defined by himself).
   - New integration of updated Diagnostic descriptions by integrator.
2. Return of completely or partly-filled DiagnosticExtract to OEM for:
   - Documentation

- ECU testing
- Integration reviews
- Failure correction

Use case examples:

- Configuration of UDS service 0x22 (`ReadDataByIdentifier`)
- Configuration of UDS service 0x2E (`WriteDataByIdentifier`)
- Configuration of UDS service 0x31 (`RoutineControl`)
- Configuration of UDS service 0x2F (`I/O-Control`)
- Configuration of DEM DTCs
- Configuration of Combined Events
- Mapping of events to DTCs
- Configuration of DTC-related environmental data
- Mapping of DEM Events to their corresponding Enable Conditions and Storage Conditions
- Configure general DCM parameters
- Description of diagnostic demands that are not relevant for code generation but have to be exchanged between OEM and Tier1 (e.g. set and reset condition for a DTC)

Refinement of use cases:

The OEM already provides a System Template for an ECU which describes the ECU Supplier SW parts as a `CompositionSwComponentType` where only inputs and outputs are known.

1. The OEM creates a `DiagnosticExtract` which describes the diagnostic interfaces of an ECU. Supported services are described (e.g. RDBI/WDBI/Routine Control) as well as their corresponding input/output parameters and return values (those which are optional in UDS standard).
2. SWC Developer at OEM or OEM SW Supplier develops SWC and also describes the Diagnostic information using `ServiceNeeds` as diagnostic contribution.
3. The OEM Diagnose Responsible for a Project creates the mappings between the `DiagnosticExtract` and the SWC available on OEM side (from 2.).
4. The OEM Diagnostic Responsible for a Project creates the mappings between the `DiagnosticExtract` and the `CompositionSwComponentType`s which will be implemented by the ECU Supplier or SW Developer.
5. The ECU Suppliers receives the ECU Extract including `DiagnosticExtract` from the OEM and imports it to the project.

6. In the same way is in point 2., the SWC Developer on supplier or Tier 2 side describes the Diagnostic information using `ServiceNeeds` as diagnostic contribution.

In the same way as in point 3: The ECU Supplier Diagnostic Responsible creates the mapping between the `PortInterface`s of the `DiagnosticExtract` (from 5.) and the `SwComponentType`s as provided in 6.

For the usage of indicators, it might happen that indicators defined on BSW level in DEM might not be automatically mapped to the implementation on SWC level. This would then require a manual mapping step by the integrator to resolve the mismatch.

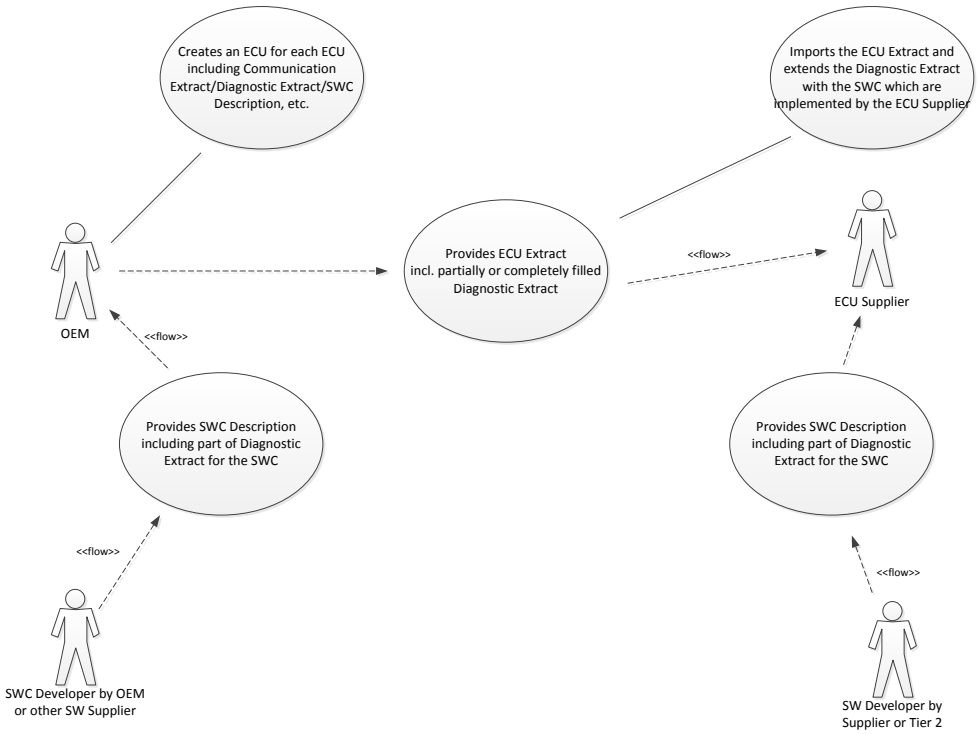

Figure 2.3: Workflow scenario for diagnostic data exchange

3 Conceptual Background

Chapter 1 has already given an overview on the intended way of using the DiagnosticExtract template and files. This chapter gives further background information on the overall concept behind the new format to create a better basis for understanding of the meta-model described in Chapter 4.

3.1 Definition of relevant Diagnostic Elements

[TPS_DEXT_01046] ECU configuration is not suitable to be exchanged between partners in an ECU development project ⌈ The ECU configuration (EcuC) parameters defined by the AUTOSAR Software Specification (SWS) documents for Dem and Dcm are not suitable to be exchanged between partners in an ECU development project.

Besides proprietary ways of using the EcuC format, the main reason for EcuC parameters being inappropriate to be exchanged is their closeness to implementation (e.g. parameter on buffer sizes). ⌋*(RS_DEXT_00001)*

[TPS_DEXT_01047] Differences in the development processes for diagnostics at automotive OEMs and ECU suppliers ⌈ Additionally, there are differences in the development processes for diagnostics at automotive OEMs and ECU suppliers resulting in different views on relevant diagnostic properties to be exchanged and different ways of deriving and defining them as diagnostic requirements. ⌋*(RS_DEXT_00002)*

Therefore, the identification of all diagnostic properties and requirements as superset from the companies' views forms the basis on which the DiagnosticExtract template has been defined.

3.2 Abstraction from EcuC Level

The DiagnosticExtract template does not only focus on relevant diagnostic properties and requirements but also - if required - lift them onto an appropriate abstraction level to make them meaningfully exchangeable (e.g. debouncing requirements that abstract from mapping on a concrete ECU).

However, for many EcuC parameters identified as relevant, an abstraction is not useful or not required and thus, these parameters are mapped 1:1 to equivalent elements of the DiagnosticExtract template.

3.3 Independence of Definition

With respect to development processes, the DiagnosticExtract format also enables more independence when defining requirements on diagnostic functionality than possible with EcuC parameters. The approach of "decentralized configuration" is met in the DiagnosticExtract template in mainly two ways described in the following sub-chapters.

3.3.1 Use of ≪atpSplitable≫ enabling separation of elements over several physical files

Most elements of the DiagnosticExtract template can be split over several physical files. Therefore, parts of these elements (e.g. certain attributes) can be defined by, for example, an OEM and other parts of these elements by, for example, an ECU supplier.

3.3.2 Use of self-contained mapping elements

Many diagnostic requirements are established by mappings between diagnostic elements (e.g., DTC to DemEvent mapping). However, the "decentralized configuration" approach requires that these mappings can be flexibly defined at almost any time within the ECU development process and by any of the involved companies respectively roles.

Therefore, the DiagnosticExtract template defines self-contained mapping elements that have references to two (or potentially more) diagnostic elements to define a mapping.

The self-contained mapping elements can be created any time after the diagnostic elements to be mapped together have been defined. Alternatively, a mapping element can be created after only one diagnostic element has been defined indicating the need to make the mapping complete by defining the additional diagnostic element(s) to map to.

4 Common Meta Model Elements

4.1 Introduction

This chapter contains a description of the meta-model for the specification of the DiagnosticExtract in AUTOSAR. The goal of the specification of the DiagnosticExtract is to contribute to the description of the configuration of the Diagnostic Communication Manager [11] (Dcm) and the Diagnostic Event Manager [12] (Dem)

The meta-model can be roughly divided into three sections

- A common section that contains meta-classes shared between the description of the diagnostic services (that roughly corresponds to the Dcm) and the diagnostic event handling (that roughly corresponds to the Dem), see section 4.2.
- A section dedicated to the configuration of the diagnostic services, see section 5.
- A section dedicated to the configuration of the diagnostic event handling, see section 6.

4.2 Data Identifier vs. Routine vs. Data Element

This chapter highlights the formal modeling of some of the central parts of AUTOSAR diagnostics when it comes to configuration. There are some concepts widely used that need to be reflected in the meta-model.

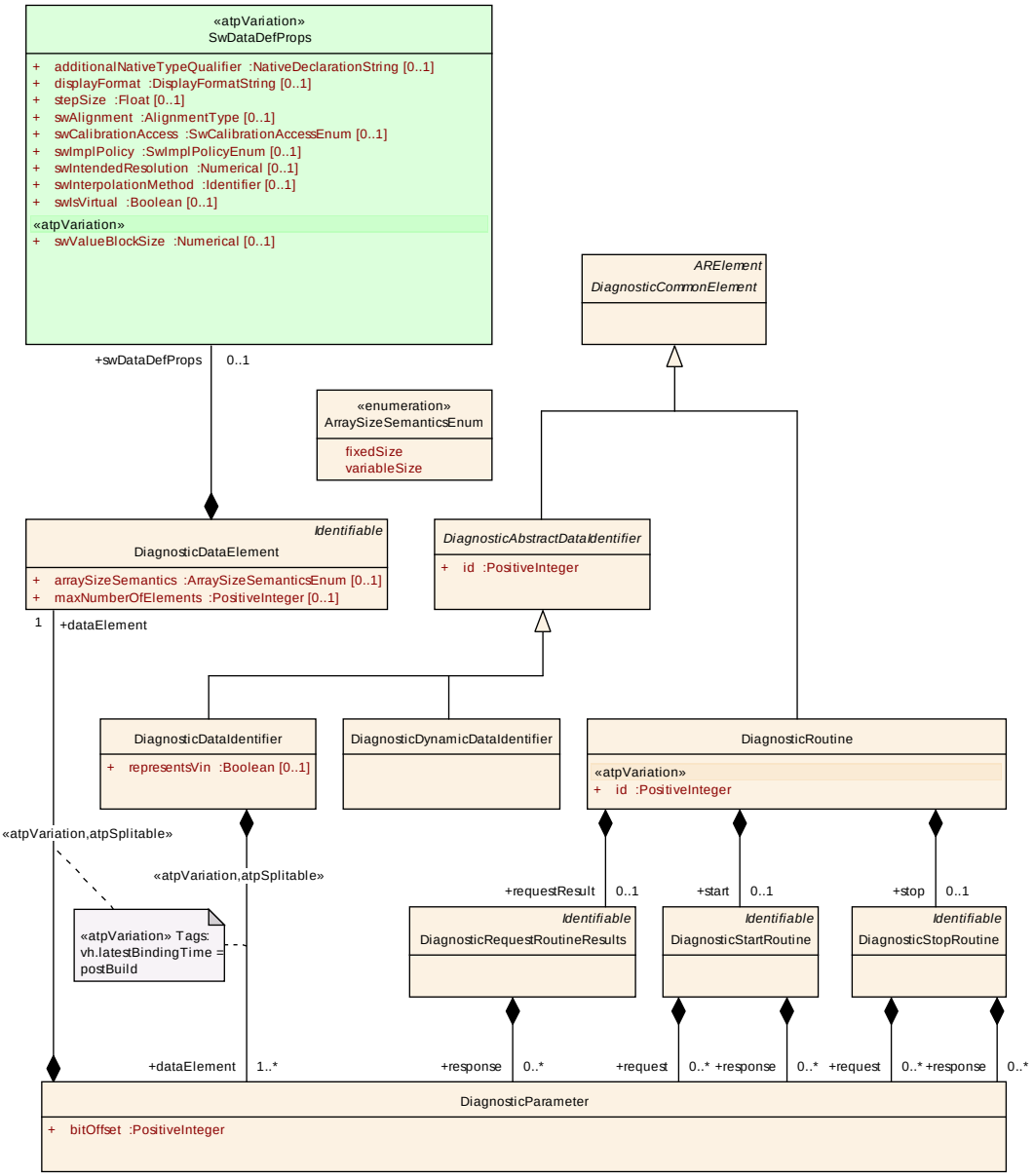

Figure 4.1: Common Diagnostic elements

Class	**DiagnosticCommonElement (abstract)**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::CommonDiagnostics			
Note	This meta-class represents a common base class for all diagnostic elements. It does not contribute any specific functionality other than the ability to become the target of a reference.			
Base	ARElement,ARObject,CollectableElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
–	–	–	–	–

Table 4.1: DiagnosticCommonElement

The purpose of the DiagnosticCommonElement is to provide a common reference target for all kinds of diagnostic elements. This aspect is explained in more detail in section 4.4.

The purpose of a Data Identifier (DID) is to associate a unique numerical value to a piece of data related to diagnostics. From the modeling point of view, this means that the modeling of the Data Identifier needs to provide an attribute that represents the numeric value as well as a relation to a Data Element representing a set of diagnostic piece of data.

[TPS_DEXT_01000] AUTOSAR diagnostics supports two kinds of data identifiers
⌈ In AUTOSAR, two kinds of data identifiers are supported:

- The DiagnosticDataIdentifier inherits from DiagnosticAbstractDataIdentifier and is used to define data identifiers fully known at configuration time. A DiagnosticDataIdentifier shall have at least 1 dataElement.
- The DiagnosticDynamicDataIdentifier inherits from DiagnosticAbstractDataIdentifier and is used to define data identifiers fully known **only** at run time. Consequently, there is **no formal means** to define dataElement at configuration time.

⌋(*RS_DEXT_00034, RS_DEXT_00035*)

Class	**DiagnosticDataIdentifier**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::CommonDiagnostics			
Note	This meta-class represents the ability to model a diagnostic data identifier (DID) that is fully specified regarding the payload at configuration-time.			
Base	ARElement,ARObject,CollectableElement,DiagnosticAbstractDataIdentifier,DiagnosticCommonElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
dataElement	DiagnosticParameter	1..*	aggr	This is the dataElement associated with the DiagnosticDataIdentifier. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=shortName, variationPoint.shortLabel vh.latestBindingTime=postBuild
representsVin	Boolean	0..1	attr	This attributes indicates whether the specific DiagnosticDataIdentifier represents the vehicle identification.

Table 4.2: DiagnosticDataIdentifier

Class	**DiagnosticDynamicDataIdentifier**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::CommonDiagnostics			
Note	This meta-class represents the ability to define a diagnostic data identifier (DID) at run-time.			
Base	ARElement,ARObject,CollectableElement,DiagnosticAbstractDataIdentifier,DiagnosticCommonElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
–	–	–	–	–

Table 4.3: DiagnosticDynamicDataIdentifier

Class	**DiagnosticAbstractDataIdentifier (abstract)**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::CommonDiagnostics			
Note	This meta-class represents an abstract base class for the modeling of a diagnostic data identifier (DID).			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
id	PositiveInteger	1	attr	This is the numerical identifier used to identify the DiagnosticAbstractDataIdentifier in the scope of diagnostic workflow

Table 4.4: DiagnosticAbstractDataIdentifier

[TPS_DEXT_01072] Purpose of attribute `DiagnosticDataIdentifier.representsVin` ⌈ There is a use case for identifying a specific `DiagnosticDataIdentifier` that carries the so-called *vehicle identification number* (VIN).

It is therefore important to be able to formally indicate this characteristic. For this purpose the attribute `DiagnosticDataIdentifier.representsVin` is available. ⌋*(RS_DEXT_00034)*

[constr_1324] Existence of attribute `DiagnosticDataIdentifier.representsVin` ⌈ Within the context of a given `DiagnosticContributionSet`, the attribute `DiagnosticDataIdentifier.representsVin` shall have the value `true` for only a single `DiagnosticDataIdentifier`. ⌋*()*

Please note that the VIN is only relevant in the context of *diagnostics over IP* (DoIP). However, there is no constraint that bounds the validity of [constr_1324] to the existence of a `DiagnosticConnection` that is build on top of an IP stack.

If the attribute exists and there is no IP used then the meaning of the attribute is simply irrelevant. Anyway, this situation should not be attributed to a misconfiguration.

The concept of the `Data Element` represents a piece of information decomposed from the data identified by a DID and exchanged between the DEM and, for example, a tester.

The nature of such a Data Element could be compared to the nature of an ISignal[1] and therefore the modeling of a Data Element by means of the meta-class DiagnosticDataElement aggregates SwDataDefProps in the role swDataDefProps in order to provide a reference to SwBaseType.

The aggregation of SwDataDefProps can also be used to refer to a DataConstr in order to specify a valid data range for the DiagnosticDataElement.

Class	**DiagnosticDataElement**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::CommonDiagnostics			
Note	This meta-class represents the ability to describe a concrete piece of data to be taken into account for diagnostic purposes.			
Base	ARObject,Identifiable,MultilanguageReferrable,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
arraySizeSemantics	ArraySizeSemanticsEnum	0..1	attr	This attribute controls the meaning of the value of the array size.
maxNumberOfElements	PositiveInteger	0..1	attr	The existence of this attribute turns the data instance into an array of data. The attribute determines the size of the array in terms of how many elements the array can take.
swDataDefProps	SwDataDefProps	0..1	aggr	This property allows to specify data definition properties in order to support the definition of e.g. computation formulae and data constraints.

Table 4.5: DiagnosticDataElement

[constr_1394] Value of DiagnosticDataElement.maxNumberOfElements depending on its existence ⌈ If the attribute DiagnosticDataElement.maxNumberOfElements exists then its value shall be greater than 0. ⌋*()*

Enumeration	**ArraySizeSemanticsEnum**
Package	M2::AUTOSARTemplates::CommonStructure::ImplementationDataTypes
Note	This type controls how the information about the number of elements in an ApplicationArrayDataType is to be interpreted.
Literal	***Description***
fixedSize	This means that the ApplicationArrayDataType will always have a fixed number of elements.
variableSize	This implies that the actual number of elements in the ApplicationArrayDataType might vary at run-time. The value of arraySize represents the maximum number of elements in the array.

Table 4.6: ArraySizeSemanticsEnum

[1] which represents the payload in "regular" bus communication

Please note that the definitions of properties like computation methods[2], limits[3], or units[4] of diagnostic data elements is based on shared resources of the AUTOSAR meta-model, namely by aggregation of meta-class SwDataDefProps.

This meta-class contributes a wealth of possible properties related to the definition of data in general and, in this case, diagnostics in particular.

However, it is important to understand that SwDataDefProps is so expressive and versatile that its applicability needs to be constrained (in this specific case, see [constr_1325]) for specific deployments according to the requirements that originate from the semantics of the piece of data that is decorated by SwDataDefProps.

[constr_1325] Allowed attributes of SwDataDefProps for DiagnosticDataElement.swDataDefProps ⌈ The allowed attributes of SwDataDefProps for the aggregation in the role DiagnosticDataElement.swDataDefProps are defined in table 4.7. ⌋*()*

Attributes of SwDataDefProps	DiagnosticDataElement.swDataDefProps
additionalNativeTypeQualifier	N/A
annotation	N/A
baseType.baseTypeDefinition.baseTypeEncoding	D
baseType.baseTypeDefinition.baseTypeSize	D
baseType.baseTypeDefinition.byteOrder	D
baseType.baseTypeDefinition.maxBaseTypeSize	N/A
baseType.baseTypeDefinition.memAlignment	N/A
baseType.baseTypeDefinition.nativeDeclaration	N/A
compuMethod	D
dataConstr	D
displayFormat	D
implementationDataType	N/A
invalidValue	N/A
mcFunction	N/A
swAddrMethod	N/A
swAlignment	N/A
swBitRepresentation	N/A
swCalibrationAccess	N/A
swCalprmAxisSet	N/A
swCalprmAxisSet.swCalprmAxis /SwAxisGrouped.swCalprmRef	N/A
swCalprmAxisSet.swCalprmAxis /SwAxisIndividual.swVariableRef	N/A
swCalprmAxisSet.swCalprmAxis /SwAxisGrouped.sharedAxisType	N/A
swCalprmAxisSet.swCalprmAxis /SwAxisIndividual.inputVariableType	N/A
swCalprmAxisSet/SwAxisIndividual.unit	N/A
swCalprmAxisSet.swCalprmAxis.baseType	N/A
swComparisonVariable	N/A
swDataDependency	N/A
swHostVariable	N/A

[2]formalized as CompuMethod in AUTOSAR
[3]formalized as DataConstr in AUTOSAR
[4]formalized as Unit in AUTOSAR

Attributes of SwDataDefProps	DiagnosticDataElement.swDataDefProps
swImplPolicy	N/A
swIntendedResolution	N/A
swInterpolationMethod	N/A
swIsVirtual	N/A
swPointerTargetProps	N/A
swRecordLayout	N/A
swRefreshTiming	N/A
swTextProps	N/A
swValueBlockSize	N/A
unit	D
valueAxisDataType	N/A

Table 4.7: Allowed attributes of SwDataDefProps for DiagnosticDataElement.swDataDefProps

The following legend applies to table 4.7:

Abbr.	Description
D	**Define** the attribute independent from settings to the left.
I	**Inherit** the definition from the left for usage in the scope of this element.
N/A	Attribute is **not applicable** for usage in the scope of this element.
M	Attribute is **meaningless** in the scope of this element. As it was allowed in previous versions, declaring it as Not Applicable (NA) would break compatibility. Tools shall ignore such an attribute without a warning.

Table 4.8: Legend of table 4.7

Please note that, in comparison to similar tables appearing in other AUTOSAR documents (e.g. [9]), table 4.7 intentionally goes into more detail regarding the applicability of the attributes of SwBaseType. This is in contrast to similar tables contained in, e.g. the specification of the Software-Component Template [9]

The attributes of SwBaseType are considered of **paramount importance for the definition of the semantics** of the enclosing DiagnosticDataElement and thus the emphasis is justified.

There are several use cases for a DiagnosticDataElement that actually represents an array of information. In some cases the array size is static and will not change at run-time, and in some cases the array size needs to change at run-time to fulfill the intended purpose.

[TPS_DEXT_01001] Definition of a fixed-sized array ⌈ A DiagnosticDataElement shall be interpreted as a **fixed-size array** if **all** of the following conditions apply:

1. The attribute DiagnosticDataElement.maxNumberOfElements exists.
2. The value of the attribute DiagnosticDataElement.maxNumberOfElements is set to a value > 0.
3. The value of DiagnosticDataElement.arraySizeSemantics either does not exist or is set to ArraySizeSemanticsEnum.fixedSize.

⌋(*RS_DEXT_00034*, *RS_DEXT_00038*)

[TPS_DEXT_01002] Definition of a variable-sized array ⌈ A `DiagnosticDataElement` shall be interpreted as a **variable-size array** if **all** of the following conditions apply:

1. The attribute `DiagnosticDataElement.maxNumberOfElements` exists.
2. The value of the attribute `DiagnosticDataElement.maxNumberOfElements` is set to a value > 0.
3. The value of `DiagnosticDataElement.arraySizeSemantics` is set to `ArraySizeSemanticsEnum.variableSize`.

The value of `DiagnosticDataElement.maxNumberOfElements` shall be considered the maximum array size in terms of the number of elements. ⌋(*RS_DEXT_00034*, *RS_DEXT_00038*)

[constr_1326] Existence of a variable-sized array ⌈ The value of the attribute `DiagnosticDataElement.arraySizeSemantics` **shall not** be set to `ArraySizeSemanticsEnum.variableSize` if the respective `DiagnosticDataElement` is referenced from a `DiagnosticServiceDataMapping`. ⌋()

4.3 Textual Documentation

A `Data Identifier` also usually comes with some textual description that explains the meaning of the `Data Identifier` in short form. This ability is available via the inheritance from `Identifiable`, in particular by means of the attributes `desc` and/or `introduction` (see Figure 4.2).

This also means that the ability to add some form of textual description is widely usable in the scope of the `DiagnosticExtract`. Many meta-classes are derived from e.g. `DiagnosticCommonElement` (which inherits from `Identifiable`) or directly from `Identifiable` and therefore qualify for the described form of documentation.

In other words, the technology described in Figures 4.2 and 4.3 is not limited to `DiagnosticDataElement` but has a much wider applicability in the context of the `DiagnosticExtract`.

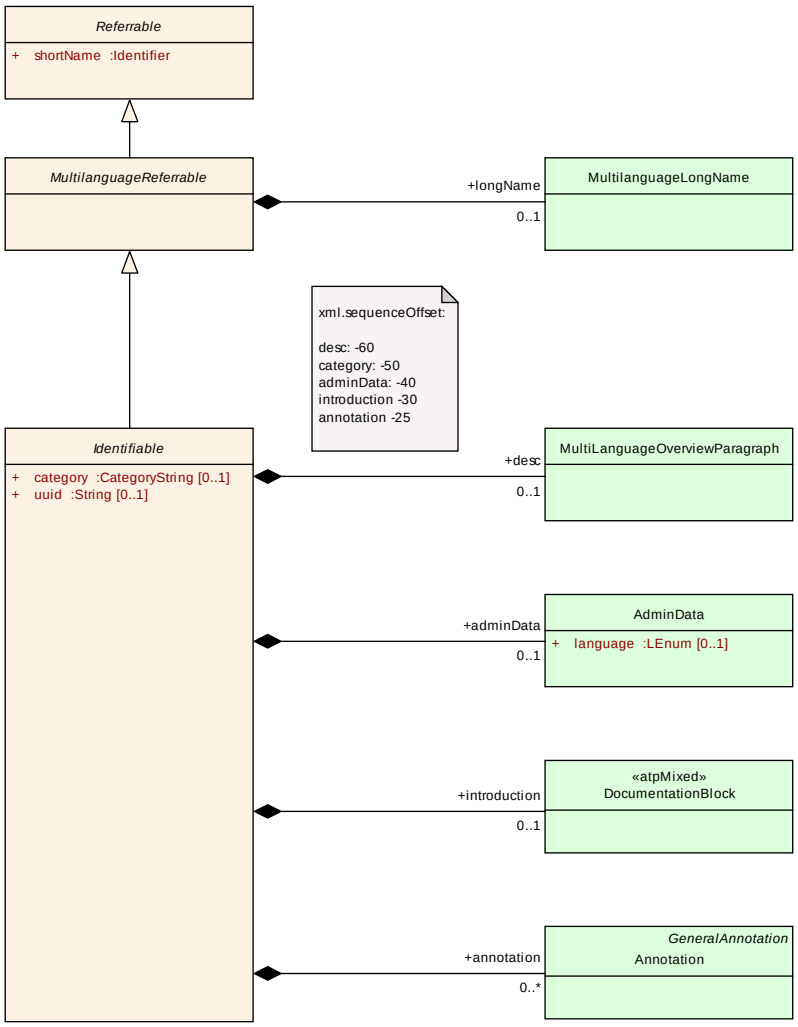

Figure 4.2: Definition of a textual description by means of `desc` and `introduction`

The details regarding the specification of textual content that goes along a given diagnostics element is detailed in Figure 4.3. In fact, `DocumentationBlock` provides a very sophisticated ability to define structured text that may consist e.g. of multiple paragraphs (formalized by meta-class `MultiLanguageOverviewParagraph` aggregated in the role `p`).

In addition to the ability to attach structured text, it is also possible to use the `annotation` (see Figure 4.2) to add short annotations (comparable to the usage of sticky notes) to diagnostic elements.

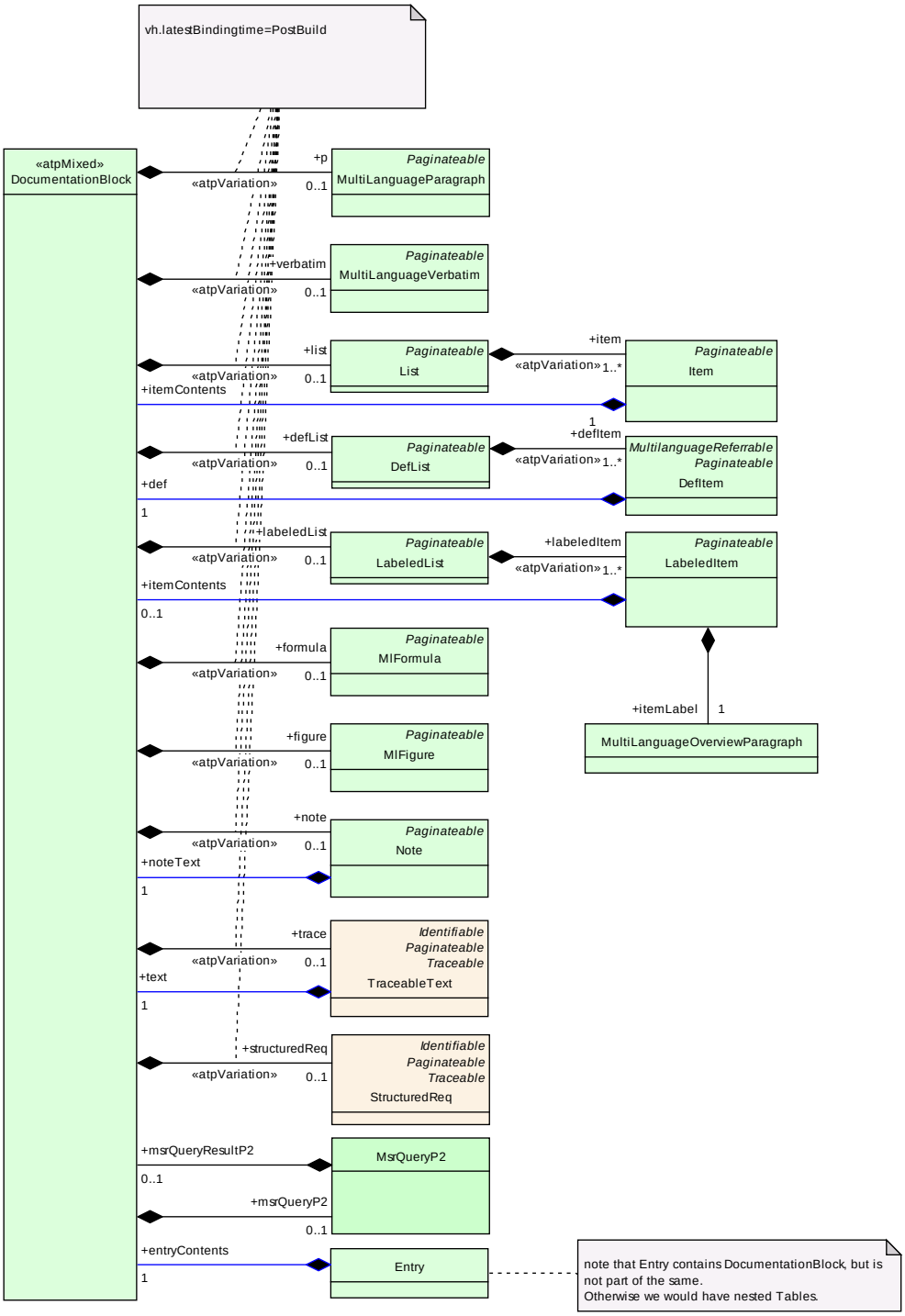

Figure 4.3: Details of the modeling of `DocumentationBlock`

4.4 Diagnostic Contribution

[TPS_DEXT_01003] DiagnosticContributionSet is the central part of the DiagnosticExtract ⌈ The central part of formalization of the concept of the DiagnosticExtract is the DiagnosticContributionSet.

To some extent, it fulfills a similar role as the System [6] in the communication domain. ⌋*(RS_DEXT_00001, RS_DEXT_00002)*

The DiagnosticContributionSet maintains references to DiagnosticCommonElement and by this means the actual definition of the extent of diagnostic contribution takes place.

In other words, the actual extent of a given contribution is created by the collection of DiagnosticCommonElements referenced by the DiagnosticContributionSet.

[TPS_DEXT_01004] DiagnosticContributionSet defines the scope of the DiagnosticExtract ⌈ The DiagnosticContributionSet has the ability to define the scope of the given DiagnosticExtract. This means that the DiagnosticContributionSet represents the DiagnosticExtract for the rest of the AUTOSAR model.

The scope may vary between the scope of an entire system down to the contribution of a specific tier-1 supplier to a much bigger context. ⌋*(RS_DEXT_00001, RS_DEXT_00002)*

[TPS_DEXT_01055] Standardized values of DiagnosticContributionSet.category ⌈ The scope of the DiagnosticContributionSet, on the other hand is determined by the value of its category. The following values are predefined by AUTOSAR:

- DIAGNOSTICS_ABSTRACT_SYSTEM_DESCRIPTION: this DiagnosticContributionSet represents a more or less high-level definition that can be taken as a template for creating concrete DiagnosticContributionSets of category DIAGNOSTICS_SYSTEM_EXTRACT or DIAGNOSTICS_ECU_EXTRACT
- DIAGNOSTICS_SYSTEM_EXTRACT: the scope of this DiagnosticContributionSet consists of several EcuInstances.
- DIAGNOSTICS_ECU_EXTRACT: the scope of this DiagnosticContributionSet consists of a **single** EcuInstances.

⌋*(RS_DEXT_00001, RS_DEXT_00002)*

[constr_1327] Multiplicity of DiagnosticContributionSet.ecuInstance ⌈ The multiplicity of DiagnosticContributionSet.ecuInstance shall be limited to 1 if the enclosing DiagnosticContributionSet is of category DIAGNOSTICS_ECU_EXTRACT. ⌋*()*

[constr_1328] Consistency of DiagnosticContributionSet.ecuInstance and DiagnosticServiceTable.ecuInstance ⌈ Each DiagnosticServiceTable referenced by any given DiagnosticContributionSet in the role serviceTable

shall define a reference in the role DiagnosticServiceTable.ecuInstance to an EcuInstance that is also referenced in the role DiagnosticContributionSet.ecuInstance by the mentioned DiagnosticContributionSet. ⌋()

Please note that [constr_1328] resolves an intentional redundancy in the meta-model. Both DiagnosticContributionSet and DiagnosticServiceTable are able refer to EcuInstance with the idea that both DiagnosticContributionSet and DiagnosticServiceTable can be modeled independently from each other.

Of course, once the DiagnosticContributionSet and DiagnosticServiceTable are integrated in the same context (in particular by establishing the reference DiagnosticContributionSet.serviceTable) the individual references to the applicable EcuInstances need to be consistent.

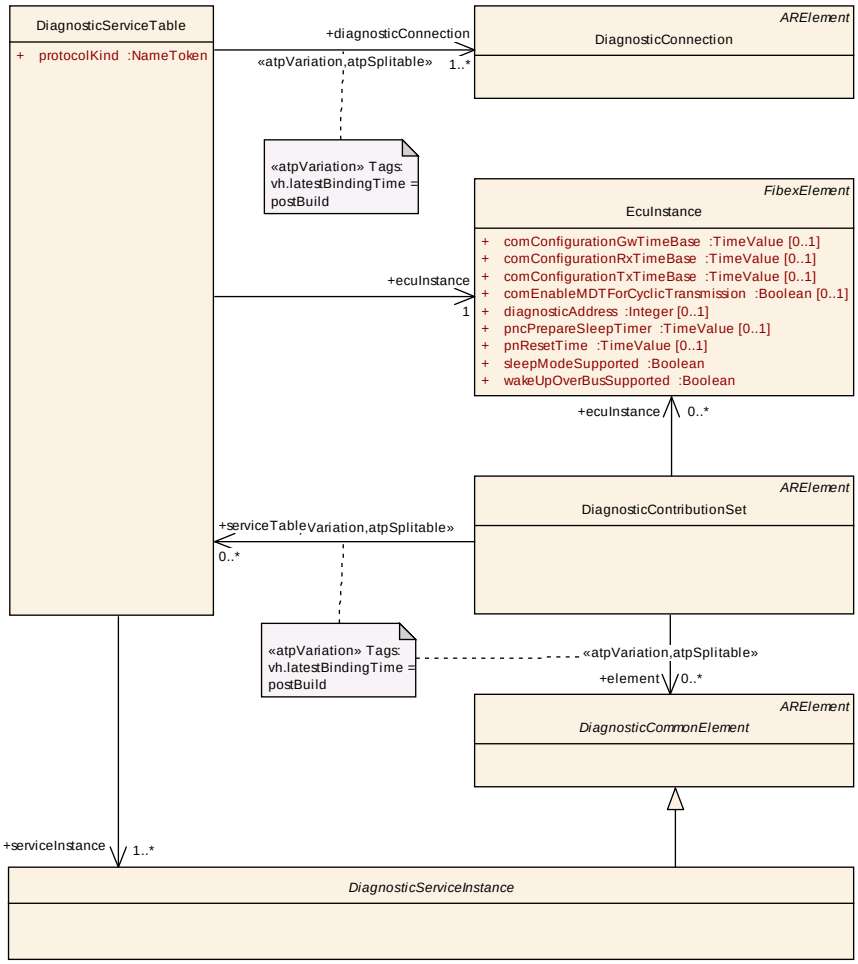

Figure 4.4: Diagnostic Contribution

[TPS_DEXT_01005] DiagnosticContributionSet can exist independently ⌈ The DiagnosticContributionSet has been modeled as an ARElement so that its instances can exist independently from the existence of context-providing model-elements inside a given ARPackage. ⌋(*RS_DEXT_00001, RS_DEXT_00002*)

[TPS_DEXT_01005] elaborates on an important aspect that makes the DiagnosticExtract independent from the existence of a context. For example, it would

have been possible to aggregate DiagnosticContributionSet somewhere, e.g. at System.

This kind of modeling intentionally puts DiagnosticContributionSet on the same level as e.g. System, as far as model granularity is concerned.

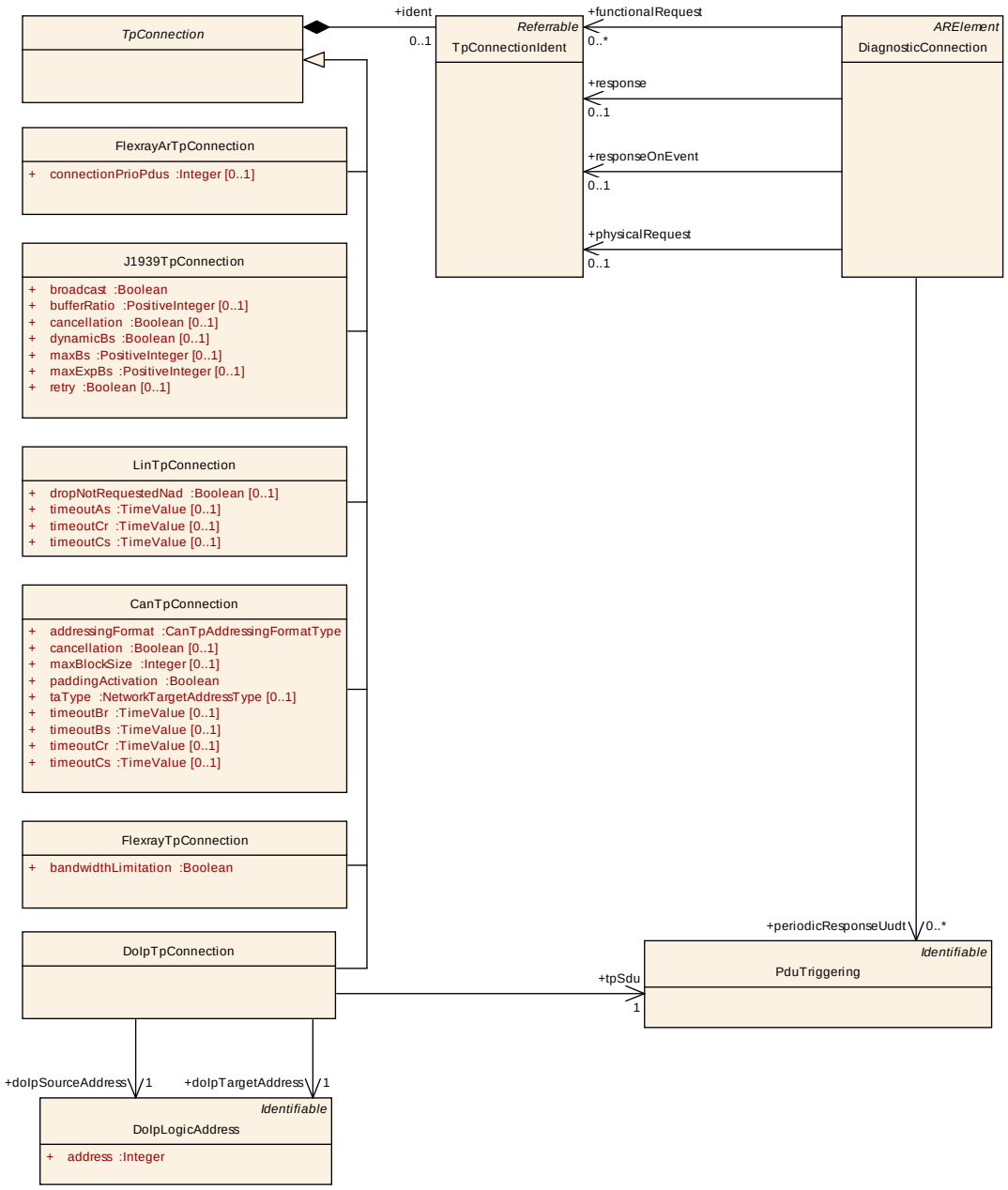

Figure 4.5: Modeling of DiagnosticConnection

[TPS_DEXT_01006] The role of DiagnosticServiceTables in the context of a DiagnosticContributionSet ⌈ The DiagnosticContributionSet also refers to a collection of DiagnosticServiceTables. These formalize the communication-protocol-related part of the DiagnosticExtract.

In other words, the service table creates a formal relation between a collection of DiagnosticServiceInstances and the DiagnosticConnection that formalizes a *conduit* for specific pairs of diagnostic request and response messages taken to transmit the diagnostic service invocations from a tester to the applicable instance of the AUTOSAR diagnostic stack and convey the response of the diagnostic stack back to the tester.

This means, that a service table describes the set of diagnostic services that are available via Diagnostic Connection which is finally a request message to address a diagnostic service to an ECU and a response message to be used by the ECU to respond to the service. ⌋*(RS_DEXT_00039)*

Here is an example of a service table for UDS diagnostics:

$14 - GroupOfDTC: 0xFFFFFF

$19 - Subfunction: $02, Subfunction $06

$22 - DataID: 0x1111, DataID: 0x2222

$2E - DataID: 0x1111, DataID: 0x2222

$2F - IO-ID:0x3333

Class	**DiagnosticContributionSet**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::DiagnosticContribution			
Note				
Base	ARElement,ARObject,CollectableElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
commonProperties	DiagnosticCommonProps	0..1	aggr	**Stereotypes:** atpSplitable**Tags:** atp.Splitkey=commonProperties
ecuInstance	EcuInstance	*	ref	This represents the collection of EcuInstances that are affected by the DiagnosticContributionSet.
element	DiagnosticCommonElement	*	ref	This represents a DiagnosticCommonElement considered in the context of the DiagnosticContributionSet **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=element, variationPoint.shortLabel vh.latestBindingTime=postBuild
serviceTable	DiagnosticServiceTable	*	ref	This represents the collection of DiagnosticServiceTables to be considered in the scope of this DiagnosticContributionSet. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=serviceTable, variationPoint.shortLabel vh.latestBindingTime=postBuild

Table 4.9: DiagnosticContributionSet

Class	**DiagnosticServiceTable**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::DiagnosticContribution			
Note				
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	*Datatype*	*Mul.*	*Kind*	*Note*
diagnosticConnection	DiagnosticConnection	1..*	ref	This represents the DiagnosticConnection that is taken for handling the data transmission for the enclosing DiagnosticServiceTable. It is possible to refer to more than one diagnosticConnections in order to support more than one diagnostic tester. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=diagnosticConnection, variationPoint.shortLabel vh.latestBindingTime=postBuild
ecuInstance	EcuInstance	1	ref	This represents the applicable EcuInstance for this DiagnosticServiceTable.
protocolKind	NameToken	1	attr	This identifies the applicable protocol.
serviceInstance	DiagnosticServiceInstance	1..*	ref	This represents the collection of DiagnosticServiceInstances to be considered in the scope of this DiagnosticServiceTable,

Table 4.10: DiagnosticServiceTable

Class	**DiagnosticConnection**			
Package	M2::AUTOSARTemplates::SystemTemplate::DiagnosticConnection			
Note	DiagnosticConncection that is used to describe the relationship between several TP connections.			
Base	ARElement,ARObject,CollectableElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	*Datatype*	*Mul.*	*Kind*	*Note*
functionalRequest	TpConnectionIdent	*	ref	Reference to functional request messages.
periodicResponseUudt	PduTriggering	*	ref	Reference to UUDT responses.
physicalRequest	TpConnectionIdent	0..1	ref	Reference to a physical request message.
response	TpConnectionIdent	0..1	ref	In the vast majority of cases a response is required. However, there are also cases where providing the response is not possible and/or not allowed.
responseOnEvent	TpConnectionIdent	0..1	ref	Reference to a ROE message.

Table 4.11: DiagnosticConnection

4.5 Diagnostic Common Properties

[TPS_DEXT_01007] Common properties of a `DiagnosticExtract` ⌈ There are some properties of a `DiagnosticExtract` that are shared among all elements of the `DiagnosticExtract`. These properties are modeled by means of the meta-class `DiagnosticCommonProps`. ⌋*(RS_DEXT_00001)*

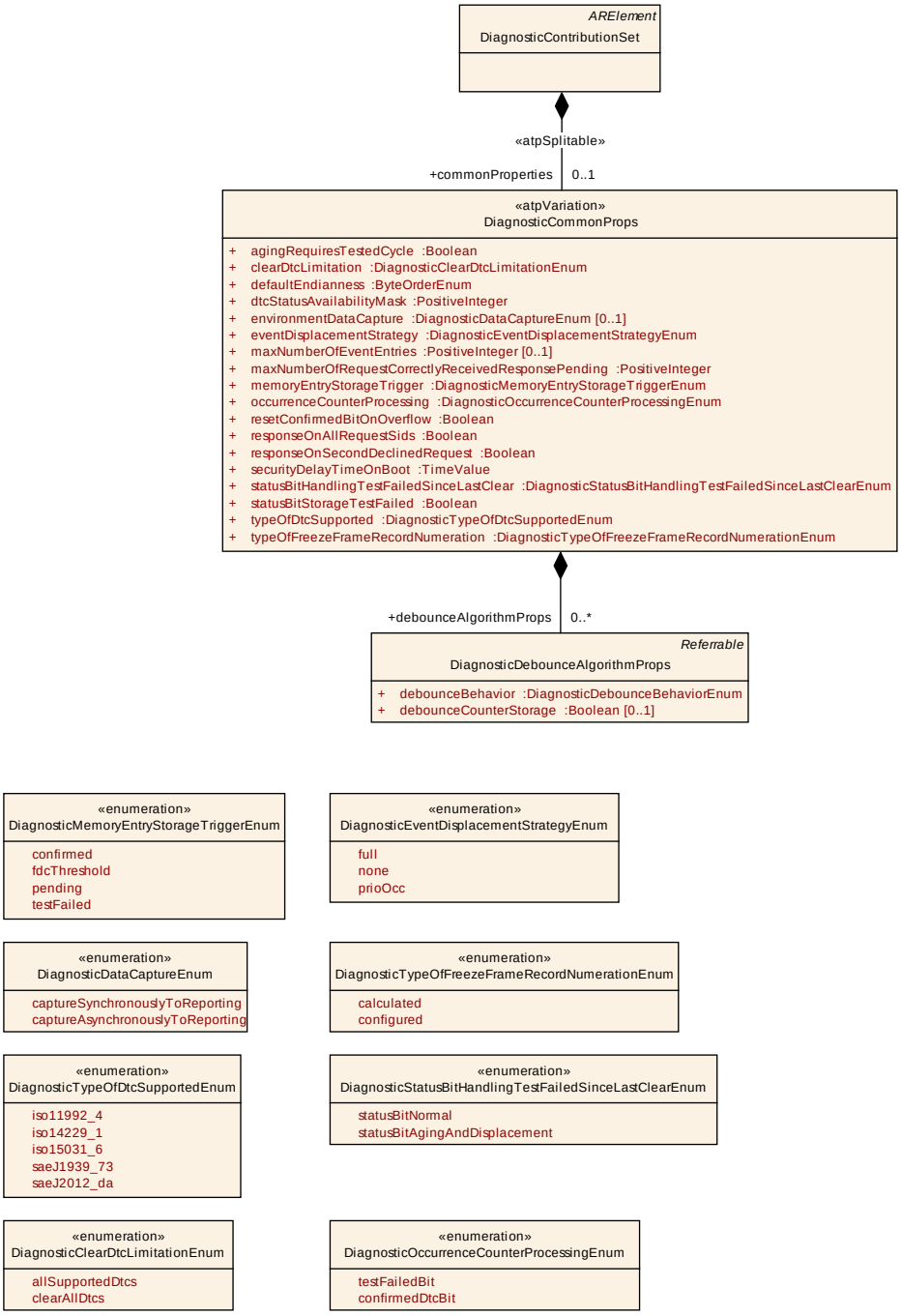

Figure 4.6: Common Diagnostic Properties

[TPS_DEXT_01008] `DiagnosticContributionSet` defines the scope for the application of the common diagnostic properties ⌈ `DiagnosticContributionSet`

aggregates DiagnosticCommonProps and by this means defines the scope for the application of the common diagnostic properties. ⌋(*RS_DEXT_00001*)

Class	≪atpVariation≫ **DiagnosticCommonProps**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::DiagnosticCommonProps			
Note	This meta-class aggregates a number of common properties that are shared among a diagnostic extract. **Tags:** vh.latestBindingTime=codeGenerationTime			
Base	ARObject			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
agingRequiresTestedCycle	Boolean	1	attr	Defines whether the aging cycle counter is processed every aging cycles or else only tested aging cycle are considered. If the attribute is set to TRUE: only tested aging cycle are considered for aging cycle counter. If the attribute is set to FALSE: aging cycle counter is processed every aging cycle.
clearDtcLimitation	DiagnosticClearDtcLimitationEnum	1	attr	Defines the scope of the DEM_ClearDTC Api.
debounceAlgorithmProps	DiagnosticDebounceAlgorithmProps	*	aggr	Defines the used debounce algorithms relevant in the context of the enclosing DiagnosticCommonProps. Usually, there is a variety of debouncing algorithms to take into account and therefore the multiplicity of this aggregation is set to 0..*.
defaultEndianness	ByteOrderEnum	1	attr	Defines the default endianness of the data belonging to a DID or RID which is applicable if the DiagnosticDataElement does not define the endianness via the swDataDefProps.baseType attribute.
dtcStatusAvailabilityMask	PositiveInteger	1	attr	Mask for the supported DTC status bits by the Dem.
environmentDataCapture	DiagnosticDataCaptureEnum	0..1	attr	This attribute determines whether the capturing of environment data is done synchronously inside the report API function or whether the capturing shall be done asynchronously, i.e. after the report API function already terminated.
eventDisplacementStrategy	DiagnosticEventDisplacementStrategyEnum	1	attr	This attribute defines, whether support for event displacement is enabled or not, and which displacement strategy is followed.
maxNumberOfEventEntries	PositiveInteger	0..1	attr	This attribute fixes the maximum number of event entries in the fault memory.

Attribute	Datatype	Mul.	Kind	Note
maxNumberOfRequestCorrectlyReceivedResponsePending	PositiveInteger	1	attr	Maximum number of negative responses with response code 0x78 (requestCorrectlyReceived-ResponsePending) allowed per request. DCM will send a negative response with response code 0x10 (generalReject), in case the limit value gets reached. Value 0xFF means that no limit number of NRC 0x78 response apply.
memoryEntryStorageTrigger	DiagnosticMemoryEntryStorageTriggerEnum	1	attr	Describes the primary trigger to allocate an event memory entry.
occurrenceCounterProcessing	DiagnosticOccurrenceCounterProcessingEnum	1	attr	This attribute defines the consideration of the fault confirmation process for the occurrence counter.
resetConfirmedBitOnOverflow	Boolean	1	attr	This attribute defines, whether the confirmed bit is reset or not while an event memory entry will be displaced.
responseOnAllRequestSids	Boolean	1	attr	If set to FALSE the DCM will not respond to diagnostic request that contains a service ID which is in the range from 0x40 to 0x7F or in the range from 0xC0 to 0xFF (Response IDs).
responseOnSecondDeclinedRequest	Boolean	1	attr	Defines the reaction upon a second request (ClientB) that can not be processed (e.g. due to priority assessment). TRUE: when the second request (Client B) can not be processed, it shall be answered with NRC21 BusyRepeatRequest. FALSE: when the second request (Client B) can not be processed, it shall not be responded.
securityDelayTimeOnBoot	TimeValue	1	attr	Start delay timer on power on in seconds. This delay indicates the time at ECU boot power-on time where the Dcm remains in the default session and does not accept a security access.
statusBitHandlingTestFailedSinceLastClear	DiagnosticStatusBitHandlingTestFailedSinceLastClearEnum	1	attr	This attribute defines, whether the aging and displacement mechanism shall be applied to the "TestFailedSinceLastClear" status bits.
statusBitStorageTestFailed	Boolean	1	attr	This parameter is used to activate/deactivate the permanent storage of the "TestFailed" status bits. true: storage activated false: storage deactivated
typeOfDtcSupported	DiagnosticTypeOfDtcSupportedEnum	1	attr	This attribute defines the format returned by Dem_DcmGetTranslationType and does not relate to/influence the supported Dem functionality.
typeOfFreezeFrameRecordNumeration	DiagnosticTypeOfFreezeFrameRecordNumerationEnum	1	attr	This attribute defines the type of assigning freeze frame record numbers for event-specific freeze frame records.

Table 4.12: DiagnosticCommonProps

Enumeration	**DiagnosticMemoryEntryStorageTriggerEnum**
Package	M2::AUTOSARTemplates::DiagnosticExtract::DiagnosticCommonProps
Note	Trigger types to allocate an event memory entry.
Literal	***Description***
confirmed	Status information of UDS DTC status bit 3
fdcThreshold	Threshold to allocate an event memory entry and to capture the Freeze Frame.
pending	Status information of UDS DTC status bit 2.
testFailed	Status information of UDS DTC status bit 0.

Table 4.13: DiagnosticMemoryEntryStorageTriggerEnum

Enumeration	**DiagnosticDataCaptureEnum**
Package	M2::AUTOSARTemplates::DiagnosticExtract::DiagnosticCommonProps
Note	Data triggering types
Literal	***Description***
captureAsynchronouslyToReporting	This represents the intention to capture the environment data asynchronously after the actual capture API function terminated.
captureSynchronouslyToReporting	This represents the intention to capture the environment data synchronously within the capture API function.

Table 4.14: DiagnosticDataCaptureEnum

Enumeration	**DiagnosticTypeOfFreezeFrameRecordNumerationEnum**
Package	M2::AUTOSARTemplates::DiagnosticExtract::DiagnosticCommonProps
Note	FreezeFrame record numeration type
Literal	***Description***
calculated	Freeze frame records will be numbered consecutive starting by 1 in their chronological order.
configured	Freeze frame records will be numbered based on the given configuration in their chronological order.

Table 4.15: DiagnosticTypeOfFreezeFrameRecordNumerationEnum

Enumeration	**DiagnosticClearDtcLimitationEnum**
Package	M2::AUTOSARTemplates::DiagnosticExtract::DiagnosticCommonProps
Note	Scope of the DEM_ClearDTC Api.
Literal	***Description***
allSupportedDtcs	DEM_ClearDtc API accepts all supported DTC values.
clearAllDtcs	DEM_ClearDtc API accepts ClearAllDTCs only.

Table 4.16: DiagnosticClearDtcLimitationEnum

Enumeration	**DiagnosticEventDisplacementStrategyEnum**

Package	M2::AUTOSARTemplates::DiagnosticExtract::DiagnosticCommonProps
Note	Defines the displacement strategy.
Literal	***Description***
full	Event memory entry displacement is enabled, by consideration of priority active/passive status, and occurrence.
none	Event memory entry displacement is disabled.
prioOcc	Event memory entry displacement is enabled, by consideration of priority and occurrence (but without active/passive status).

Table 4.17: DiagnosticEventDisplacementStrategyEnum

Enumeration	**DiagnosticOccurrenceCounterProcessingEnum**
Package	M2::AUTOSARTemplates::DiagnosticExtract::DiagnosticCommonProps
Note	The occurrence counter triggering types.
Literal	***Description***
confirmedDtcBit	The occurrence counter is triggered by the TestFailed bit if the fault confirmation was successful (ConfirmedDTC bit is set).
testFailedBit	The occurrence counter is only triggered by the TestFailed bit (and the fault confirmation is not considered).

Table 4.18: DiagnosticOccurrenceCounterProcessingEnum

Enumeration	**DiagnosticTypeOfDtcSupportedEnum**
Package	M2::AUTOSARTemplates::DiagnosticExtract::DiagnosticCommonProps
Note	Supported Dtc Types
Literal	***Description***
iso11992_4	ISO11992-4 DTC format
iso14229_1	ISO14229-1 DTC format (3 byte format)
iso15031_6	ISO15031-6 DTC format (2 byte format)
saeJ1939_73	SAEJ1939-73 DTC format
saeJ2012_da	SAE_J2012-DA_DTCFormat_00 (3 byte format)

Table 4.19: DiagnosticTypeOfDtcSupportedEnum

In addition to the already described common diagnostic properties there are further properties that are specific to an individual `EcuInstance`.

[TPS_DEXT_01073] Diagnostic properties that are specific to an individual `EcuInstance` ⌈ Diagnostic properties that are specific to an individual `EcuInstance` are modeled by means of the meta-class `DiagnosticEcuProps` that is aggregated at `EcuInstance` in the role `diagnosticProps`. ⌋*(RS_DEXT_00048)*

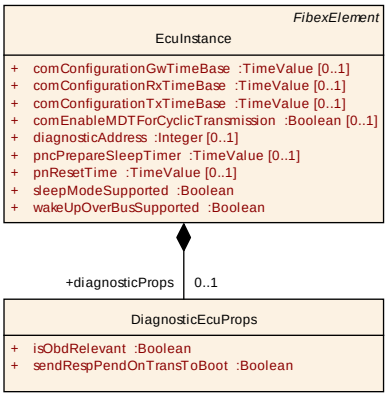

Figure 4.7: ECU-specific diagnostic properties

Class	**DiagnosticEcuProps**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::DiagnosticCommonProps			
Note	This meta-class is defined to gather diagnostic-related properties that apply in the scope of an entire ECU.			
Base	ARObject			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
isObdRelevant	Boolean	1	attr	This attribute indicates whether the ECU makes any contribution to the OBD.
sendRespPendOnTransToBoot	Boolean	1	attr	The purpose of this attribute is to define whether or not the ECU should send a NRC 0x78 (response pending) before transitioning to the bootloader (in this case the attribute shall be set to "true") or if the transition shall be initiated without sending NRC 0x78 (in this case the attribute shall be set to "false").

Table 4.20: DiagnosticEcuProps

5 Diagnostic Services

5.1 Introduction

The meta-model for the diagnostic services according to AUTOSAR, to a large degree, takes over aspects of the description of diagnostic services according to the definition of *Unified Diagnostic Services* (UDS) as of ISO 14229 [15].

5.2 Service Instance vs. Service Class

When it comes to diagnostic services, the meta-model distinguishes between the concept of a diagnostic service *instance* vs. the concept of a diagnostic service *class*.

As the terminology suggests, the diagnostic service *instance* (formalized as DiagnosticServiceInstance) implements a concrete use of a diagnostic service in a given context whereas purpose of the diagnostic service *class* (formalized as DiagnosticServiceClass) is to provide properties that are shared among all existing diagnostic service *instance*s in the model.

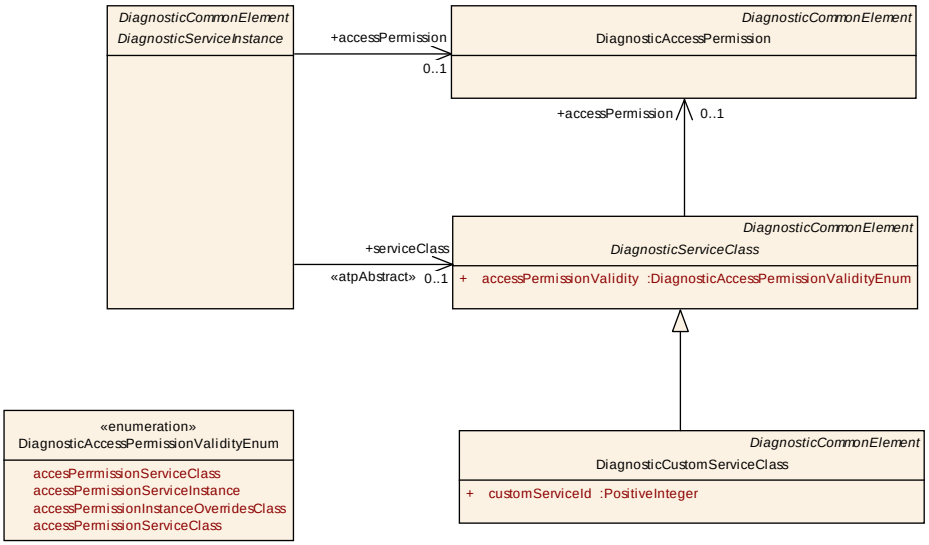

Figure 5.1: Common service elements

[constr_1329] Existence of concrete sub-classes of DiagnosticServiceClass in the context created by a DiagnosticContributionSet ⌈ One of the following mutually exclusive conditions shall apply for the existence of any concrete sub-class of DiagnosticServiceClass in the context created by a DiagnosticContributionSet:

- The subclass of DiagnosticServiceClass shall only appear once in the context created by a DiagnosticContributionSet
- If the subclass of DiagnosticServiceClass appears multiple times in the context created by a DiagnosticContributionSet then all instances shall have identical values for all of their attributes.

  In case of aggregations the number of aggregated elements shall be identical and the values of primitive attributes of aggregated elements shall again be identical.

⌋()

The background of [constr_1329] is obviously related to the semantics of DiagnosticServiceClass which is to define model attributes that are shared among all DiagnosticServiceInstances.

This would not be possible if more that one DiagnosticServiceClass with a diverging set of attribute values exists.

Class	**DiagnosticServiceClass (abstract)**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::CommonService			
Note	This meta-class provides the ability to define common properties that are shared among all instances of sub-classes of DiagnosticServiceInstance.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
accessPermission	DiagnosticAccessPermission	0..1	ref	This represents the collection of DiagnosticAccessPermissions that allow for the execution of the referencing DiagnosticServiceClass.
accessPermissionValidity	DiagnosticAccessPermissionValidityEnum	1	attr	This attribute is responsible for clarifying the validity of the accessPermission reference.

Table 5.1: DiagnosticServiceClass

Class	**DiagnosticServiceInstance (abstract)**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::CommonService			
Note	This represents a concrete instance of a diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
accessPermission	DiagnosticAccessPermission	0..1	ref	This represents the collection of DiagnosticAccessPermissions that allow for the execution of the referencing DiagnosticServiceInstance..
serviceClass	DiagnosticServiceClass	0..1	ref	This represents the corresponding "class", i.e. this meta-class provides properties that are shared among all instances of applicable sub-classes of DiagnosticServiceInstance. The subclasses that affected by this pattern implement references to the applicable "class"-role that substantiate this abstract reference. **Stereotypes:** atpAbstract

Table 5.2: DiagnosticServiceInstance

[TPS_DEXT_01009] Limited support for the configuration of custom diagnostic services ⌈ Beside the support for explicitly modeled diagnostic services, there is also a limited support for the configuration of custom diagnostic services.

The formalization, however, goes only so far as to define the `DiagnosticCustomServiceClass` and its attribute `customServiceId` that allows for the definition of the custom service identifier used for the purpose. ⌋*(RS_DEXT_00047)*

[constr_1330] Custom service identifier shall not overlap with standardized service identifiers ⌈ The value of the attribute customServiceId shall not be set to any of the values reserved for standardized service identifiers as defined by the ISO 14229-1, see [15]. ⌋*()*

[TPS_DEXT_01010] Configuration of custom diagnostic services ⌈ The support for the configuration of custom diagnostic services within the DiagnosticExtract does not extend beyond the ability to define that attribute DiagnosticCustomServiceClass.customServiceId.

There is no corresponding formalization of a diagnostic service instance that supports references to e.g. a DiagnosticDataIdentifier. ⌋*(RS_DEXT_00047)*

Class	**DiagnosticCustomServiceClass**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::CommonService			
Note	This represents the ability to define a custom diagnostic service class and assign an ID to it. Further configuration is not foreseen from the point of view of the diagnostic extract and consequently needs to be done on the level of ECUC.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticServiceClass,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
customServiceId	PositiveInteger	1	attr	This attribute may only be used for the definition of custom services. The values shall not overlap with existing standardized service IDs.

Table 5.3: DiagnosticCustomServiceClass

5.3 Access Permission, Session, Security Level

This chapter discusses a set of meta-classes that have been created to represent the concept of an *access permission* used in the context of the Dcm.

5.3.1 Introduction to Access Permission

The DiagnosticAccessPermission is used to describe the ability (or the lack thereof) to execute a diagnostic service depending on the referenced DiagnosticSecurityLevel and DiagnosticSession (see Figure 5.2).

At runtime, DiagnosticSessions are used to create a context for the execution of diagnostic functionality. Servers usually support a variety of different DiagnosticSessions. It is possible to switch between DiagnosticSessions at runtime.

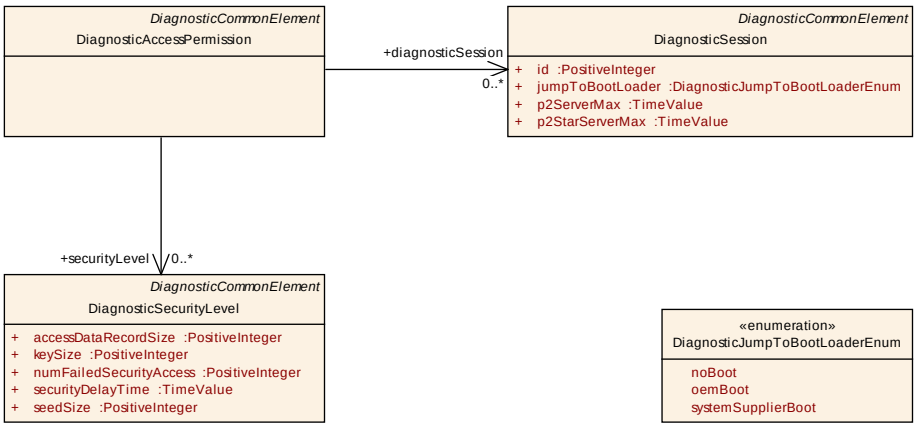

Figure 5.2: Common model elements relevant for the Dcm

[TPS_DEXT_01011] Semantics of `DiagnosticSession.id` ⌈ The value of the attribute `DiagnosticSession.id` has a given semantics according to ISO 14229-1 [15]. For the sake of completeness, the dedicated values of `DiagnosticSession.id` are:

0x01 This represents the *default session*. This session has a specific semantics in the context of diagnostics communication such that e.g. any diagnostic service that is supposed to execute in the *default session* cannot require any reference to a `DiagnosticSecurityLevel`.

0x02 This represents the *programming session*.

0x03 This represents the *extended diagnostic session*.

0x04 This represents the *safety system diagnostic session*.

The value range **0x40** .. **0x5F** is reserved for manufacturer-specific use. ⌋*(RS_DEXT_00040)*

[TPS_DEXT_01012] Rationale for the modeling of the multiplicity of `DiagnosticAccessPermission.securityLevel` ⌈ The multiplicity of `DiagnosticAccessPermission.securityLevel` has been set to 0..* with the following motivation:

- The `DiagnosticSession` where the attribute `DiagnosticSession.id` is set to 0x01 shall **not** be associated with a `DiagnosticSecurityLevel`.
- There are no associated `DiagnosticSecurityLevel`s required. As a consequence, the execution of the `DiagnosticServiceInstance` that references the given `DiagnosticAccessPermission` is always possible.

⌋*(RS_DEXT_00041, RS_DEXT_00042)*

[TPS_DEXT_01070] Description of textually semi-formal formulated pre- and run-conditions for the validity of `DiagnosticAccessPermission` ⌈ AUTOSAR supports the description of textually formulated semi-formal pre- and run-conditions for the validity of `DiagnosticAccessPermission`.

This can be done by means of the attribute DiagnosticAccessPermission.introduction.structuredReq. ⌋()

For more details regarding the modeling of the semi-formal text please refer to Figure 4.3. An example of how the definition of pre- and run-conditions may look like in ARXML is sketched in listing 5.1.

To make this approach work it is important to standardize possible values of the attribute category such that the semi-formal semantics of the definition of pre- and run-conditions is protected by regulation of the AUTOSAR standard.

[TPS_DEXT_01071] Standardized values of DiagnosticAccessPermission.introduction.structuredReq ⌈ The following possible values of DiagnosticAccessPermission.introduction.struturedReq are standardized by AUTOSAR:

- **DIAG_ACCESS_PERM_PRE_COND**: this value describes the pre-condition of the corresponding DiagnosticAccessPermission.
- **DIAG_ACCESS_PERM_RUN_COND**: this value describes the run-condition of the corresponding DiagnosticAccessPermission.

⌋(*RS_DEXT_00041*, *RS_DEXT_00045*)

Listing 5.1: Example for the definition of pre- and run-conditions for DiagnosticAccessPermission

```
<DIAGNOSTIC-ACCESS-PERMISSION>
    <SHORT-NAME>exampleAccessPermission</SHORT-NAME>
    <INTRODUCTION>
        <STRUCTURED-REQ>
            <SHORT-NAME>precondition</SHORT-NAME>
            <CATEGORY>DIAG_ACCESS_PERM_PRE_COND</CATEGORY>
            <DESCRIPTION>
                <P>
                    <L-1 L="EN">This is a textual description of a pre-
                        condition</L-1>
                </P>
            </DESCRIPTION>
        </STRUCTURED-REQ>
        <STRUCTURED-REQ>
            <SHORT-NAME>runcondition</SHORT-NAME>
            <CATEGORY>DIAG_ACCESS_PERM_RUN_COND</CATEGORY>
            <DESCRIPTION>
                <P>
                    <L-1 L="EN">This is a textual description of a run-
                        condition</L-1>
                </P>
            </DESCRIPTION>
        </STRUCTURED-REQ>
    </INTRODUCTION>
    <DIAGNOSTIC-SESSION-REFS>
        <DIAGNOSTIC-SESSION-REF DEST="DIAGNOSTIC-SESSION">/AUTOSAR/
            UseCase_230/ExampleSession</DIAGNOSTIC-SESSION-REF>
    </DIAGNOSTIC-SESSION-REFS>
```

```
    <SECURITY-LEVEL-REFS>
        <SECURITY-LEVEL-REF DEST="DIAGNOSTIC-SECURITY-LEVEL">/AUTOSAR/
            UseCase_230/ExampleSecurityLevel</SECURITY-LEVEL-REF>
    </SECURITY-LEVEL-REFS>
</DIAGNOSTIC-ACCESS-PERMISSION>
```

Class	**DiagnosticAccessPermission**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm			
Note	This represents the specification of whether a given service can be accessed according to the existence of meta-classes referenced by a particular DiagnosticAccessPermission. In other words, this meta-class acts as a mapping element between several (otherwise unrelated) pieces of information that are put into context for the purpose of checking for access rights.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
diagnosticSession	DiagnosticSession	*	ref	This represents the associated DiagnosticSessions
securityLevel	DiagnosticSecurityLevel	*	ref	This represents the associated DiagnosticSecurityLevels

Table 5.4: DiagnosticAccessPermission

Class	**DiagnosticSession**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm			
Note	This meta-class represents the ability to define a diagnostic session.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
id	PositiveInteger	1	attr	This is the numerical identifier used to identify the DiagnosticSession in the scope of diagnostic workflow
jumpToBootLoader	DiagnosticJumpToBootLoaderEnum	1	attr	This attribute represents the ability to define whether this diagnostic session allows to jump to Bootloader (OEM Bootloader or System Supplier Bootloader). If this diagnostic session doesn't allow to jump to Bootloader the value JumpToBootLoaderEnum.noBoot shall be chosen.
p2ServerMax	TimeValue	1	attr	This is the session value for P2ServerMax in seconds (per Session Control). The AUTOSAR configuration standard is to use SI units, so this parameter is defined as a float value in seconds.

Attribute	*Datatype*	*Mul.*	*Kind*	*Note*
p2StarServerMax	TimeValue	1	attr	This is the session value for P2*ServerMax in seconds (per Session Control). The AUTOSAR configuration standard is to use SI units, so this parameter is defined as a float value in seconds.

Table 5.5: DiagnosticSession

Enumeration	**DiagnosticJumpToBootLoaderEnum**
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm
Note	This enumeration contains the options for jumping to a boot loader.
Literal	*Description*
noBoot	This diagnostic session doesn't allow to jump to Bootloader.
oemBoot	This diagnostic session allows to jump to OEM Bootloader.
systemSupplierBoot	This diagnostic session allows to jump to System Supplier Bootloader.

Table 5.6: DiagnosticJumpToBootLoaderEnum

Class	**DiagnosticSecurityLevel**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm			
Note	This meta-class represents the ability to define a security level considered for diagnostic purposes.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	*Datatype*	*Mul.*	*Kind*	*Note*
accessDataRecordSize	PositiveInteger	1	attr	This represents the size of the AccessDataRecord used in GetSeed. Unit:byte.
keySize	PositiveInteger	1	attr	This represents the size of the security key. Unit: byte.
numFailedSecurityAccess	PositiveInteger	1	attr	This represents the number of failed security accesses after which the delay time is activated.
securityDelayTime	TimeValue	1	attr	This represents the delay time after a failed security access. Unit: second.
seedSize	PositiveInteger	1	attr	This represents the size of the security seed. Unit: byte.

Table 5.7: DiagnosticSecurityLevel

5.3.2 Prioritization of Access Permission

The definition of *access permission* itself can be done on different levels. It is therefore necessary to define how the existence of *access permission* on these different levels is supposed to be interpreted.

[TPS_DEXT_01061] Supported scenarios for the definition of *access permission*
⌈ The following scenarios are possible for the definition of *access permission*:

- The *access permission* is defined on the level of a DiagnosticServiceClass. In this scenario, the **intended semantics** is that this configuration is binding for **all** DiagnosticServiceInstances derived from the DiagnosticServiceClass.

  The configuration of a DiagnosticServiceInstance.accessPermission is considered as an error and shall be reported accordingly.

  This scenario applies if DiagnosticServiceClass.accessPermissionValidity is set to the value accessPermissionServiceClass.

- The *access permission* is defined on the level of an individual DiagnosticServiceInstance. In this scenario, the **intended semantics** is that the DiagnosticServiceClass shall **not make any assumptions about the definition of the applicable *access permission***.

  The configuration of a DiagnosticServiceClass.accessPermission is considered as an error and shall be reported accordingly. This scenario applies if DiagnosticServiceClass.accessPermissionValidity is set to the value accessPermissionServiceInstance.

- The definition of both DiagnosticServiceClass.accessPermission and DiagnosticServiceInstance.accessPermission is positively allowed.

  In this scenario, the **intended semantics** is that if DiagnosticServiceClass.accessPermission exists the individual DiagnosticServiceInstances **are not required to define** DiagnosticServiceInstance.accessPermission **but if they do** then the DiagnosticServiceInstance.accessPermission **gets priority** over the definition of DiagnosticServiceClass.accessPermission.

  This basically boils down to the ability to **override** the setting for *access permission* made on the level of a DiagnosticServiceClass by the setting on the level of an DiagnosticServiceInstance.

  At the same time, this scenario saves some file footprint because (given the existence of DiagnosticServiceClass.accessPermission) there is no need to define individual DiagnosticServiceInstance.accessPermission unless there is a dedicated need for them.

  This scenario applies if DiagnosticServiceClass.accessPermissionValidity is set to the value accessPermissionInstanceOverridesClass.

⌋(*RS_DEXT_00041*, *RS_DEXT_00049*, *RS_DEXT_00050*)

The scenarios defined by [TPS_DEXT_01061] need modeling support in order to allow the user to precisely express the intended semantics of a model with respect to *access permission*. For this purpose the attribute `DiagnosticServiceClass.accessPermissionValidity` is available.

Enumeration	**DiagnosticAccessPermissionValidityEnum**
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::Common Service
Note	This meta-class provides settings for how the accessPermission shall be resolved between DiagnosticServiceInstance and DiagnosticServiceClass.
Literal	***Description***
accesPer-rmission ServiceClass	This means that the DiagnosticServiceClass is in charge to define the accessPermission. **Tags:** atp.Status=obsolete
accessPer-mission Instance Overrides Class	This means that accessPermission set at the DiagnosticServiceInstance will override the accessPermission defined at the DiagnosticServiceClass.
accessPer-mission ServiceClass	This means that the DiagnosticServiceClass is in charge to define the accessPermission.
accessPer-mission Service Instance	This means that the DiagnosticServiceInstance is in charge of defining the accessPermission

Table 5.8: DiagnosticAccessPermissionValidityEnum

[TPS_DEXT_01062] Existence of `DiagnosticServiceClass.accessPermissionValidity` in an incomplete model ⌈ If the attribute `DiagnosticServiceClass.accessPermissionValidity` does not exist then it shall be assumed that the configuration is incomplete. ⌋(*RS_DEXT_00001*, *RS_DEXT_00002*, *RS_DEXT_00041*)

Please note that the model state described in [TPS_DEXT_01062] is still allowed because it may only be possible to decide about the value of the attribute at later points in the model's life cycle.

[TPS_DEXT_01063] Existence of `DiagnosticServiceClass.accessPermissionValidity` in a complete model ⌈ As the model's life cycle approaches the point where the model is considered complete the attribute `DiagnosticServiceClass.accessPermissionValidity` shall exist in order to be able to properly figure out the intended model semantics. ⌋(*RS_DEXT_00001*, *RS_DEXT_00002*, *RS_DEXT_00041*)

[TPS_DEXT_01052] Existence of attribute DiagnosticServiceInstance.accessPermission ⌈ Regarding the existence of the attribute DiagnosticServiceInstance.accessPermission the following rules apply:

- If neither the attribute DiagnosticServiceInstance.accessPermission or DiagnosticServiceClass.accessPermission exist it is assumed that the configuration is incomplete as no access permission is defined.
- If either the attribute DiagnosticServiceInstance.accessPermission or DiagnosticServiceClass.accessPermission exists but does not have further references to DiagnosticSession or DiagnosticSecurityLevel then this means that the affected diagnostic services can be executed in any diagnostic session or security level. In other words, no restriction applies.

⌋(*RS_DEXT_00041*, *RS_DEXT_00049*)

5.4 Diagnostic Services supported by AUTOSAR

The following sub-chapters describe the modeling of the collection of relevant diagnostic services as defined in the ISO 14229-1 [15]. This means that the definition of the AUTOSAR DiagnosticExtract does not explicitly support the total collection of diagnostic services as defined by [15].

Some of the diagnostic services compiled in this document define so-called subfunctions that need to be identified to fully specify the nature of the respective diagnostic service.

[TPS_DEXT_01045] Supported diagnostic services ⌈ The table 5.9 shows the UDS services supported by the DiagnosticExtract. ⌋(*RS_DEXT_00003*, *RS_DEXT_00004*, *RS_DEXT_00005*, *RS_DEXT_00006*, *RS_DEXT_00007*, *RS_DEXT_00008*, *RS_DEXT_00009*, *RS_DEXT_00010*, *RS_DEXT_00011*, *RS_DEXT_00012*, *RS_DEXT_00013*, *RS_DEXT_00014*, *RS_DEXT_00015*, *RS_DEXT_00016*, *RS_DEXT_00017*, *RS_DEXT_00018*, *RS_DEXT_00019*, *RS_DEXT_00020*, *RS_DEXT_00021*, *RS_DEXT_00022*)

ID	Service
0x10	SessionControl
0x11	ECUReset
0x14	ClearDiagnosticInformation
0x19	ReadDTCInformation
0x22	ReadDataByIdentifier
0x23	ReadMemoryByAddress
0x27	SecurityAccess
0x28	CommunicationControl
0x2A	ReadDataByPeriodicIdentifier
0x2C	DynamicallyDefineDataIdentifier
0x2E	WriteDataByIdentifier
0x2F	IOControl
0x31	RoutineControl

0x34	RequestDownload
0x35	RequestUpload
0x36	TransferData
0x37	RequestTransferExit
0x3D	WriteMemoryByAddress
0x85	ControlDTCSetting
0x86	ResponseOnEvent

Table 5.9: Supported diagnostic services

[TPS_DEXT_01013] Specification of sub-functions by means of attribute DiagnosticServiceInstance.category ⌈

In all cases where a diagnostic service defines a sub-function according to ISO 14229-1 [15], the value of the attribute category of the applicable sub-class of DiagnosticServiceInstance can be used to specify the applicable sub-function as a textual token.

Constraints are defined to clarify the existence of standardized values of the attribute category for the given sub-function. This implies that an instance of the given sub-class of DiagnosticServiceInstance only has a single sub-function at a time. ⌋*(RS_DEXT_00049, RS_DEXT_00051)*

[TPS_DEXT_01014] Possible values of the category attribute for diagnostic services ⌈ AUTOSAR claims the right to standardize the possible values of the attribute category for given diagnostic services. ⌋*(RS_DEXT_00001, RS_DEXT_00051)*

If applicable, AUTOSAR allows for the usage of values of the attribute category other than the standardized values.

In this case, however, proprietary values of the attribute category shall be prefixed with a company-specific name fragment in order to avoid collisions that could occur if or when the list of possible values claimed by the AUTOSAR standard itself is extended. Example:

Listing 5.2: Example for the definition of a custom category

```
<AR-PACKAGE>
    <SHORT-NAME>DiagnosticExtractExample</SHORT-NAME>
    <ELEMENTS>
        <DIAGNOSTIC-ECU-RESET>
            <SHORT-NAME>ResetTheHardWay</SHORT-NAME>
            <CATEGORY>ACME_REALLY_HARD_RESET</CATEGORY>
        </DIAGNOSTIC-ECU-RESET>
    </ELEMENTS>
</AR-PACKAGE>
```

5.4.1 DataByIdentifier

This chapter describes the modeling of diagnostic services `ReadDataByIdentifier` (0x22) and `WriteDataByIdentifier` (0x2E).

The purpose of this diagnostic service is to enable a tester to request the values of data records from the AUTOSAR diagnostics stack. The data records are identified by a formally modeled `DiagnosticDataIdentifier`.

The modeling of this diagnostic service comprises the two meta-classes `DiagnosticReadDataByIdentifier` and `DiagnosticWriteDataByIdentifier`. These meta-classes both need to specify the set of `DiagnosticDataIdentifier`s as well as the set of applicable `DiagnosticAccessPermission`s.

As these properties are shared between instances of `DiagnosticReadDataByIdentifier` and `DiagnosticWriteDataByIdentifier`, an abstract base class named `DiagnosticDataByIdentifier` has been created that provides the actual references to `DiagnosticDataIdentifier` and `DiagnosticAccessPermission`.

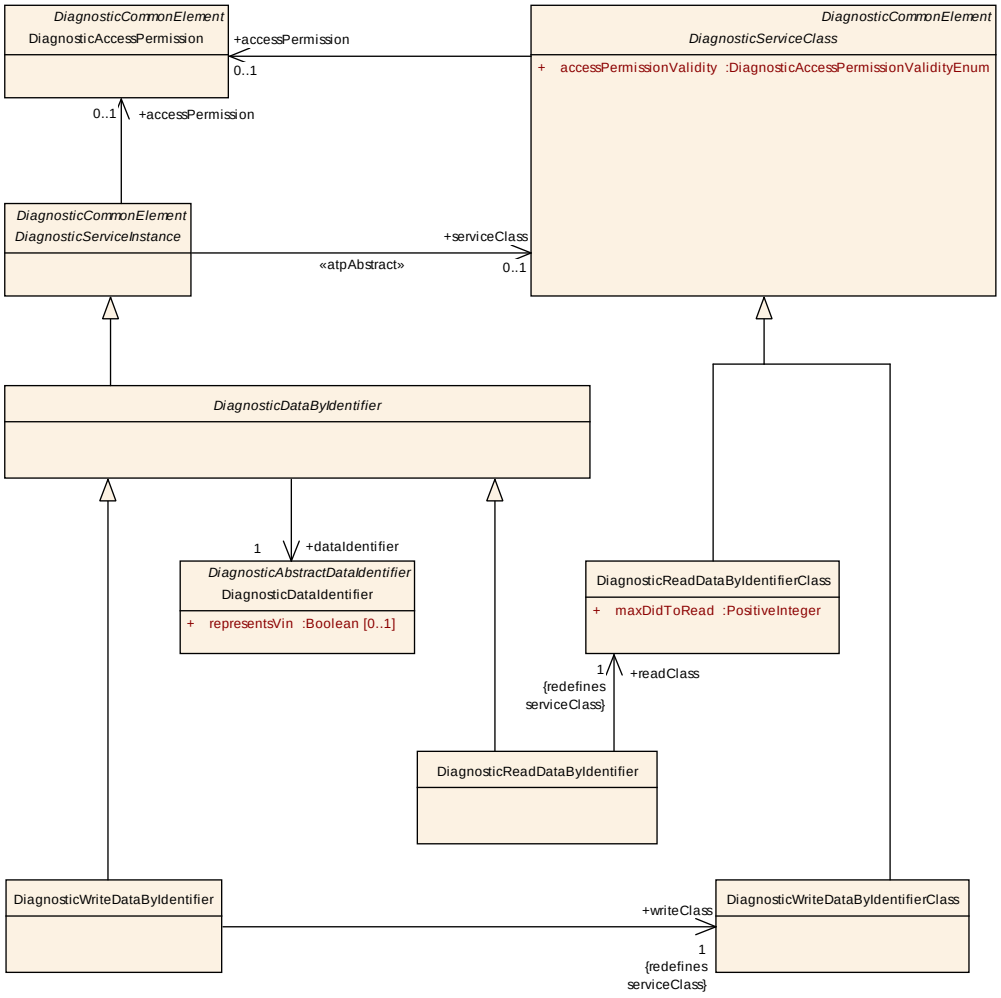

Figure 5.3: Modeling of diagnostic services `ReadDataByIdentifier` (0x22) and `WriteDataByIdentifier` (0x2E)

[TPS_DEXT_01054] Existence of `DiagnosticDataByIdentifier.dataIdentifier` ⌈ The configuration of a given `DiagnosticDataByIdentifier` is considered incomplete until the reference in the role `DiagnosticDataByIdentifier.dataIdentifier` exists. ⌋*(RS_DEXT_00007, RS_DEXT_00013, RS_DEXT_00034)*

The meaning of [TPS_DEXT_01054] is that the reference may be missing in intermediate steps of the configuration work flow. But at the point in time where ECU configuration is generated from the `DiagnosticExtract` the reference is needed to able to make sense of the model for the given `DiagnosticDataByIdentifier`.

The ability to read multiple DIDs at run-time is controlled via attribute `DiagnosticReadDataByIdentifier.maxDidToRead` and therefore it is sufficient to (at configuration-time) limit the multiplicity of attribute `dataIdentifier` to 1.

Class	**DiagnosticReadDataByIdentifier**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::DataByIdentifier			
Note	This represents an instance of the "Read Data by Identifier" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticDataByIdentifier,DiagnosticServiceInstance,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
readClass	DiagnosticReadDataByIdentifierClass	1	ref	This reference substantiates that abstract reference in the role serviceClass for this specific concrete class. Thereby, the reference represents the ability to access shared attributes among all DiagnosticReadDataByIdentifier in the given context.

Table 5.10: DiagnosticReadDataByIdentifier

Class	**DiagnosticWriteDataByIdentifier**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::DataByIdentifier			
Note	This represents an instance of the "Write Data by Identifier" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticDataByIdentifier,DiagnosticServiceInstance,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
writeClass	DiagnosticWriteDataByIdentifierClass	1	ref	This reference substantiates that abstract reference in the role serviceClass for this specific concrete class. Thereby, the reference represents the ability to access shared attributes among all DiagnosticWriteDataByIdentifier in the given context.

Table 5.11: DiagnosticWriteDataByIdentifier

Class	DiagnosticWriteDataByIdentifierClass			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::DataBy Identifier			
Note	This meta-class contains attributes shared by all instances of the "Write Data by Identifier" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Diagnostic ServiceClass,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
–	–	–	–	–

Table 5.12: DiagnosticWriteDataByIdentifierClass

Class	DiagnosticDataByIdentifier (abstract)			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::DataBy Identifier			
Note	This represents an abstract base class for all diagnostic services that access data by identifier.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Diagnostic ServiceInstance,Identifiable,MultilanguageReferrable,Packageable Element,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
dataIdentifi er	DiagnosticDataI dentifier	1	ref	This represents the linked DiagnosticDataIdentifier.

Table 5.13: DiagnosticDataByIdentifier

The modeling of `DiagnosticDataByIdentifier` represents concrete instances of diagnostic services within a `DiagnosticExtract`. However, there are attributes that are shared among all instances of `DiagnosticReadDataByIdentifier`.

For this purpose the dedicated service class `DiagnosticReadDataByIdentifierClass` has been introduced.

Class	DiagnosticReadDataByIdentifierClass			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::DataBy Identifier			
Note	This meta-class contains attributes shared by all instances of the "Read Data by Identifier" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Diagnostic ServiceClass,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
maxDidTo Read	PositiveInteger	1	attr	This attribute represents the maximum number of allowed DIDs in a single instance of DiagnosticReadDataByIdentifier.

Table 5.14: DiagnosticReadDataByIdentifierClass

Please note that it is possible to create a reference to a concrete `DiagnosticDataIdentifier` from different `DiagnosticServiceInstance`s.

[TPS_DEXT_01050] Consistency of `DiagnosticServiceSwMapping` with respect to data IDs ⌈ For each `DiagnosticServiceSwMapping` that references a `DiagnosticValueNeeds` and a `DiagnosticDataByIdentifier`, the value of `DiagnosticValueNeeds.didNumber` shall be ignored and the value of `DiagnosticDataByIdentifier.dataIdentifier.id` shall be taken instead. ⌋*(RS_DEXT_00007, RS_DEXT_00013, RS_DEXT_00034, RS_DEXT_00052)*

5.4.2 IOControl

This chapter describes the modeling of diagnostic services `InputOutput Control` (0x2F). The purpose of this service is to provide the tester with the ability to override values exchanged with the AUTOSAR hardware abstraction.

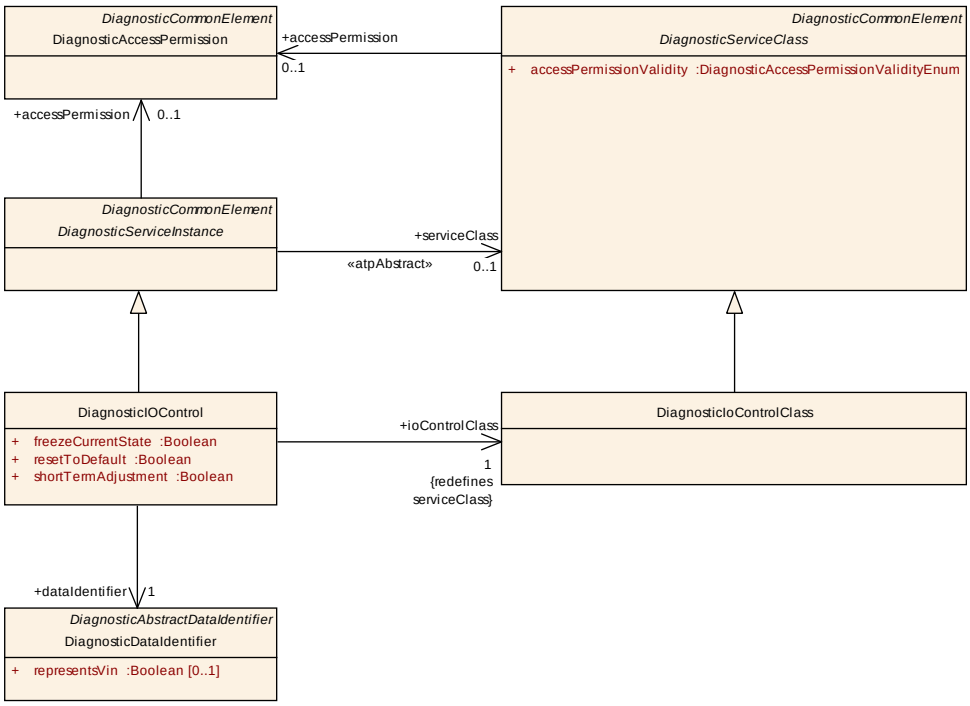

Figure 5.4: Modeling of diagnostic service `IOControl` (0x2F)

[TPS_DEXT_01015] Meaning of attributes of `DiagnosticIOControl` ⌈ The attributes `freezeCurrentState`, `resetToDefault`, and `shortTermAdjustment` represent the capabilities of the server rather than a concrete request message. ⌋*(RS_DEXT_00014)*

[TPS_DEXT_01016] The capability returnControlToEcu ⌈ According to the statement made by [TPS_DEXT_01015], there is no formal means provided to configure the capability to execute *returnControlToEcu*. This lack of configuration is intentional because the capability is always available and cannot be revoked anyway. ⌋*(RS_DEXT_00014)*

[TPS_DEXT_01017] Meaning of `DiagnosticIOControl.dataIdentifier` ⌈ The `DiagnosticIOControl.dataIdentifier` is taken for specifying the payload for the service.

However, in some cases `dataIdentifier` models the payload of the request message (`DiagnosticIOControl.shortTermAdjustment` is set to true) and in some cases it represents the payload of the response message. ⌋*(RS_DEXT_00014, RS_DEXT_00034)*

Please note that the referenced `dataIdentifier` itself may aggregate several `DiagnosticDataElement`s.

At run-time, only some `DiagnosticDataElement`s may be relevant for a specific execution of the service `InputOutput Control`. For this purpose, the diagnostic message contains the so-called `ControlEnableMaskRecord` (for more information, please refer to [SWS_DCM_00581]).

[TPS_DEXT_01089] Definition of an *identifier* of a `DiagnosticIOControl` ⌈ The *identifier* of a `DiagnosticIOControl` is defined by the attribute `DiagnosticIOControl.dataIdentifier.id`. ⌋*(RS_DEXT_00037)*

Class	**DiagnosticIOControl**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::IOControl			
Note	This represents an instance of the "I/O Control" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticServiceInstance,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
dataIdentifier	DiagnosticDataIdentifier	1	ref	This represents the corresponding DiagnosticDataIdentifier
freezeCurrentState	Boolean	1	attr	Setting this attribute to true represents the ability of the Dcm to execute a freezeCurrentState.
ioControlClass	DiagnosticIoControlClass	1	ref	This reference substantiates that abstract reference in the role serviceClass for this specific concrete class. Thereby, the reference represents the ability to access shared attributes among all DiagnosticIOControl in the given context.
resetToDefault	Boolean	1	attr	Setting this attribute to true represents the ability of the Dcm to execute a resetToDefault.
shortTermAdjustment	Boolean	1	attr	Setting this attribute to true represents the ability of the Dcm to execute a shortTermAdjustment.

Table 5.15: DiagnosticIOControl

Class	**DiagnosticIoControlClass**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::IOControl			
Note	This meta-class contains attributes shared by all instances of the "IO Control" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticServiceClass,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
–	–	–	–	–

Table 5.16: DiagnosticIoControlClass

[TPS_DEXT_01018] `InputOutput Control` does not define any sub-functions ⌈ The diagnostic service `InputOutput Control` does not define any sub-functions, therefore the value of `DiagnosticIOControl.category` does not need to be constrained. ⌋*(RS_DEXT_00014, RS_DEXT_00051)*

[TPS_DEXT_01051] Consistency of `DiagnosticServiceSwMapping` with respect to data IDs ⌈ For each `DiagnosticServiceSwMapping` that references a `DiagnosticIoControlNeeds` and a `DiagnosticIOControl`, the value of `DiagnosticIoControlNeeds.didNumber` shall be ignored and the value of `DiagnosticIOControl.dataIdentifier.id` shall be taken instead. ⌋*(RS_DEXT_00014, RS_DEXT_00052)*

5.4.3 EcuReset

This chapter describes the modeling of diagnostic services `EcuReset` (0x11).

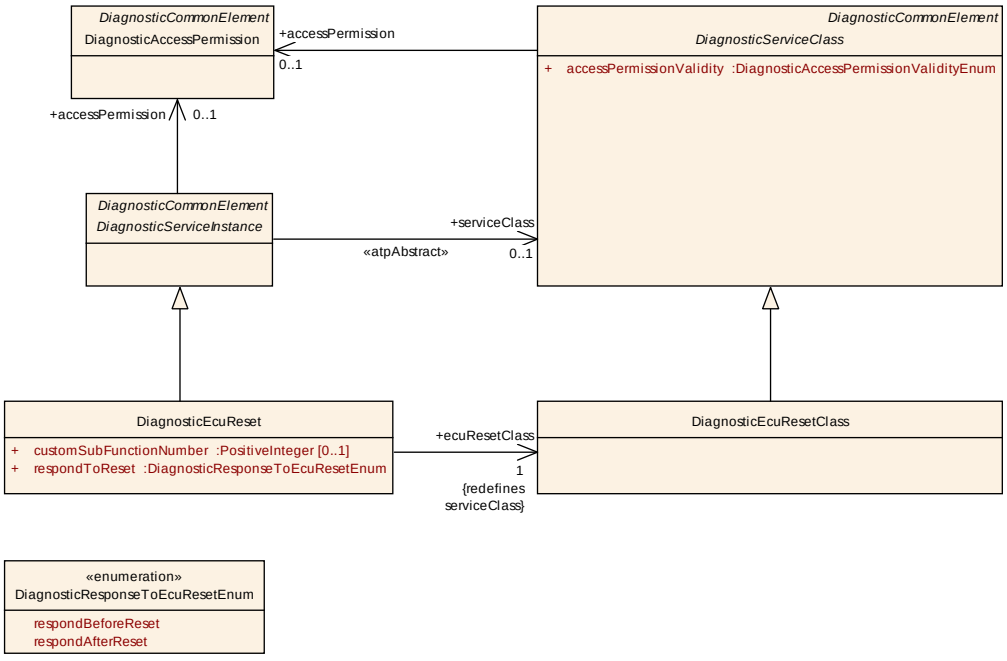

Figure 5.5: Modeling of diagnostic service `EcuReset` (0x11)

Class	**DiagnosticEcuReset**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::EcuReset			
Note	This represents an instance of the "ECU Reset" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticServiceInstance,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
customSubFunctionNumber	PositiveInteger	0..1	attr	This attribute shall be used to define a custom sub-function number if none of the standardized values of category shall be used.
ecuResetClass	DiagnosticEcuResetClass	1	ref	This reference substantiates that abstract reference in the role serviceClass for this specific concrete class. Thereby, the reference represents the ability to access shared attributes among all DiagnosticEcuReset in the given context.
respondToReset	DiagnosticResponseToEcuResetEnum	1	attr	This attribute defines whether the response to the EcuReset service shall be transmitted before or after the actual reset.

Table 5.17: DiagnosticEcuReset

Class	**DiagnosticEcuResetClass**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::EcuReset			
Note	This meta-class contains attributes shared by all instances of the "Ecu Reset" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticServiceClass,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
–	–	–	–	–

Table 5.18: DiagnosticEcuResetClass

Please note that (as already explained in section 5.4) the `SubFunctions` of this service are modeled by means of the `category` attribute.

[TPS_DEXT_01056] Applicable values for `DiagnosticEcuReset.category` ⌈
The following values of the attribute `DiagnosticEcuReset.category` are standardized by AUTOSAR:

- `HARD_RESET`
- `KEY_OFF_ON_RESET`
- `SOFT_RESET`
- `ENABLE_RAPID_POWER_SHUT_DOWN`
- `DISABLE_RAPID_POWER_SHUT_DOWN`

The meaning of these values is described in the applicable ISO document [15]. ⌋(*RS_DEXT_00001, RS_DEXT_00004, RS_DEXT_00051*)

[TPS_DEXT_01019] Correspondence of `category` values to numerical values mentioned in the ISO 14229-1 ⌈ The ISO 14229-1 [15] standard document defines specific numerical values for the sub-functions of the diagnostic service `EcuReset`.

The correspondence of the numerical values to the pre-defined values of `category` according to [TPS_DEXT_01056] is pretty obvious because the definition of values defined in [TPS_DEXT_01056] has been directly inspired by the ISO 14229-1 [15] standard document. ⌋(*RS_DEXT_00001, RS_DEXT_00004, RS_DEXT_00051*)

[TPS_DEXT_01020] Manufacturer-specific values for sub-functions of service `EcuReset` ⌈ The ISO 14229-1 [15] standard document, beyond the standardized numerical values for sub-functions, reserves a numerical range of subFunction identifiers for manufacturer- or supplier-specific use.

In this case it is possible to define further values for `category`, provided that a custom prefix is used to avoid potential name clashes with further extensions of the AUTOSAR standard, namely [TPS_DEXT_01056]. ⌋(*RS_DEXT_00001, RS_DEXT_00004, RS_DEXT_00051*)

[TPS_DEXT_01021] Semantics of `DiagnosticEcuReset.customSubFunctionNumber` ⌈ The attribute `DiagnosticEcuReset.customSubFunctionNumber` has been introduced to allow for the specification of a manufacturer- or supplier-specific value to represent the custom sub-function in the diagnostic communication.

The tuple created by the the values of attributes `DiagnosticEcuReset.category` and `DiagnosticEcuReset.customSubFunctionNumber` fully specifies identification of the manufacturer- or supplier-specific sub-function. ⌋(*RS_DEXT_00004, RS_DEXT_00047, RS_DEXT_00051*)

[constr_1331] Existence of `DiagnosticEcuReset.customSubFunctionNumber` ⌈ The attribute `DiagnosticEcuReset.customSubFunctionNumber` shall only exist if the value of `DiagnosticEcuReset.category` is outside the standardized set of values as defined by [TPS_DEXT_01056]. ⌋()

[constr_1332] Value range for `DiagnosticEcuReset.customSubfunctionNumber` ⌈ The allowed value for `DiagnosticEcuReset.customSubfunctionNumber` shall always be within the closed interval **0x40** .. **0x7E**. ⌋()

5.4.4 ClearDiagnosticInformation

This chapter describes the modeling of diagnostic services `ClearDiagnosticInformation` (0x14). As the name suggests, the purpose of the service is to clear diagnostic information in the AUTOSAR diagnostics stack.

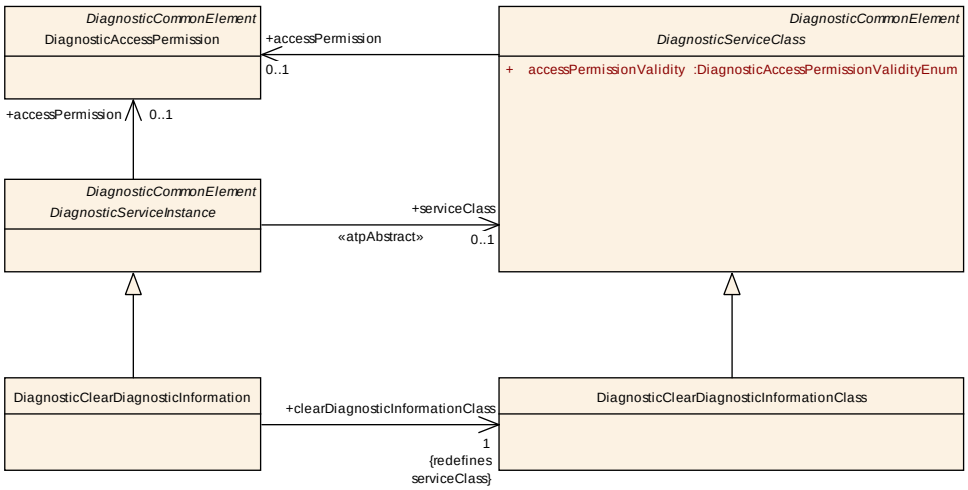

Figure 5.6: Modeling of diagnostic service `ClearDiagnosticInformation` (0x14)

Please note that there is nothing to configure for `DiagnosticClearDiagnosticInformation` beyond its mere existence.

Class	**DiagnosticClearDiagnosticInformation**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::ClearDiagnosticInfo			
Note	This represents an instance of the "Clear Diagnostic Information" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticServiceInstance,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
clearDiagnosticInformationClass	DiagnosticClearDiagnosticInformationClass	1	ref	This reference substantiates that abstract reference in the role serviceClass for this specific concrete class. Thereby, the reference represents the ability to access shared attributes among all DiagnosticClearDiagnosticInformation in the given context.

Table 5.19: DiagnosticClearDiagnosticInformation

Class	**DiagnosticClearDiagnosticInformationClass**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::ClearDiagnosticInfo			
Note	This meta-class contains attributes shared by all instances of the "Clear Diagnostic Information" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticServiceClass,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
–	–	–	–	–

Table 5.20: DiagnosticClearDiagnosticInformationClass

[TPS_DEXT_01022] `ClearDiagnosticInformation` does not define any sub-functions ⌈ The diagnostic service `ClearDiagnosticInformation` does not define any sub-functions, therefore the value of `DiagnosticClearDiagnosticInformation.category` does not need to be constrained. ⌋*(RS_DEXT_00001, RS_DEXT_00005, RS_DEXT_00051)*

5.4.5 MemoryByAddress

This chapter describes the modeling of diagnostic services for memory access (0x23, 03D, 0x34-0x37). The purpose of these services is to access memory on the diagnostic stack on request of the tester.

The service description for accessing memory for diagnostic purposes is modeled by the abstract meta-class `DiagnosticMemoryByAddress`. It is supposed to provide all model properties relevant for the memory access.

The description of memory access to some extent requires a formal description of the memory segments to take into account. For this purpose the meta-class `DiagnosticMemoryIdentifier` has been introduced and aggregated at `DiagnosticMemoryByAddress` in the role `memoryRange`.

The intent of this modeling was not to provide a generic memory model but to allow for the specification of memory properties just as far as diagnostics is concerned.

The aggregation of `DiagnosticMemoryIdentifier` at `DiagnosticMemoryByAddress` may or may not be relevant for an OEM. However, there is certainly a use case for adding this information to a `DiagnosticExtract` that goes back from a tier-1 supplier to an OEM as sort of a documentation of the diagnostic configuration.

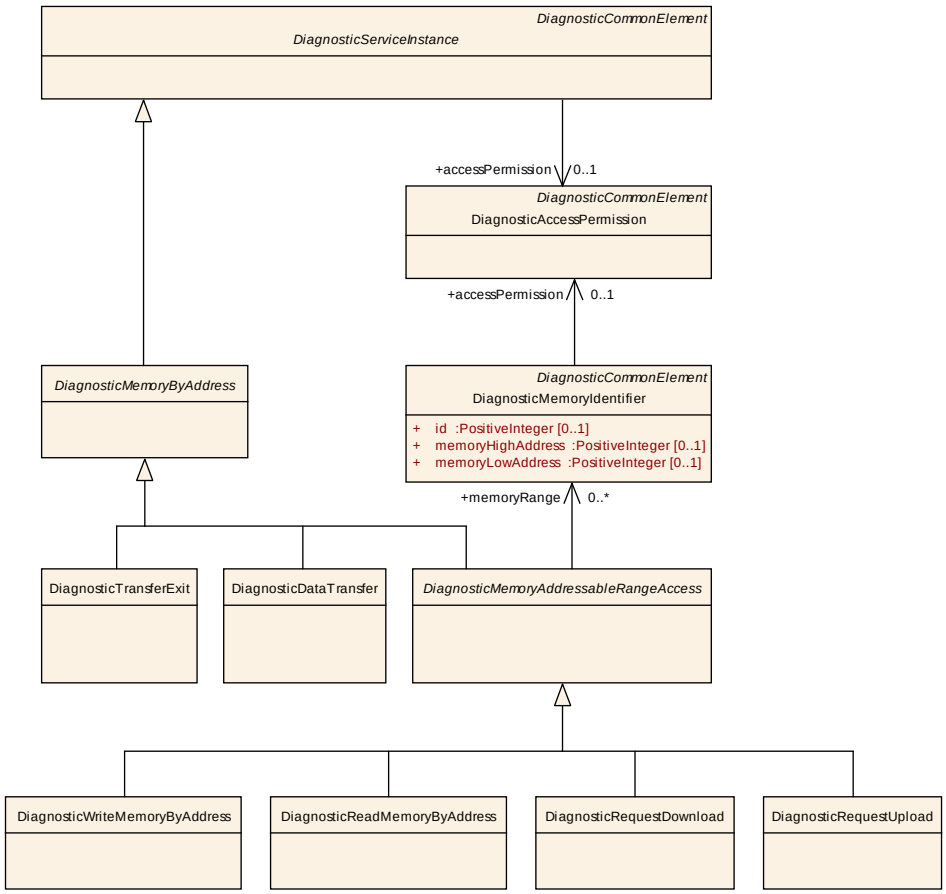

Figure 5.7: Modeling of diagnostic services `Memory` ((0x23, 03D, 0x34-0x37)

As `DiagnosticMemoryByAddress` represents a generic base class for all kinds of diagnostic memory access, it is also necessary to model the particular sub-classes that address specific use cases for diagnostic memory access.

These sub-classes are conceptually on the same level as other sub-classes of `DiagnosticServiceInstance`.

In other words, the case of memory access deviates from the modeling of other diagnostic services such that there is one further abstract base class involved.

[constr_1333] Existence of `DiagnosticMemoryIdentifier.memoryLowAddress` and `DiagnosticMemoryIdentifier.memoryHighAddress` ⌈ The attributes `DiagnosticMemoryIdentifier.memoryLowAddress` as well as `DiagnosticMemoryIdentifier.memoryHighAddress` shall not exist if the `DiagnosticMemoryIdentifier` referenced in the role `memoryRange` is referenced by a `DiagnosticRequestDownload` or a `DiagnosticRequestUpload`. ⌋*()*

Class	DiagnosticMemoryByAddress (abstract)			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::MemoryByAddress			
Note	This represents an abstract base class for diagnostic services that deal with accessing memory by address.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticServiceInstance,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
–	–	–	–	–

Table 5.21: DiagnosticMemoryByAddress

Class	DiagnosticMemoryAddressableRangeAccess (abstract)			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::MemoryByAddress			
Note	This abstract base class			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticMemoryByAddress,DiagnosticServiceInstance,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
memoryRange	DiagnosticMemoryIdentifier	*	ref	This represents the formal description of the memory segment to which the DiagnosticMemoryByAddress applies.

Table 5.22: DiagnosticMemoryAddressableRangeAccess

Class	DiagnosticMemoryIdentifier			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::MemoryByAddress			
Note	This meta-class represents the ability to define memory properties from the diagnostics point of view.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
accessPermission	DiagnosticAccessPermission	0..1	ref	This represents that access permission defined for the specific DiagnosticMemoryIdentifier.
id	PositiveInteger	0..1	attr	This represents the identification of the memory segment.
memoryHighAddress	PositiveInteger	0..1	attr	This represents the upper bound for addresses of the memory segment.
memoryLowAddress	PositiveInteger	0..1	attr	This represents the lower bound for addresses of the memory segment.

Table 5.23: DiagnosticMemoryIdentifier

Class	**DiagnosticWriteMemoryByAddress**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::MemoryByAddress			
Note	This represents an instance of the "Write Memory by Address" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticMemoryAddressableRangeAccess,DiagnosticMemoryByAddress,DiagnosticServiceInstance,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
writeClass	DiagnosticWriteMemoryByAddressClass	1	ref	This reference substantiates that abstract reference in the role serviceClass for this specific concrete class. Thereby, the reference represents the ability to access shared attributes among all DiagnosticWritememoryByAddress in the given context.

Table 5.24: DiagnosticWriteMemoryByAddress

Class	**DiagnosticWriteMemoryByAddressClass**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::MemoryByAddress			
Note	This meta-class contains attributes shared by all instances of the "Write Memory by Address" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticServiceClass,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
–	–	–	–	–

Table 5.25: DiagnosticWriteMemoryByAddressClass

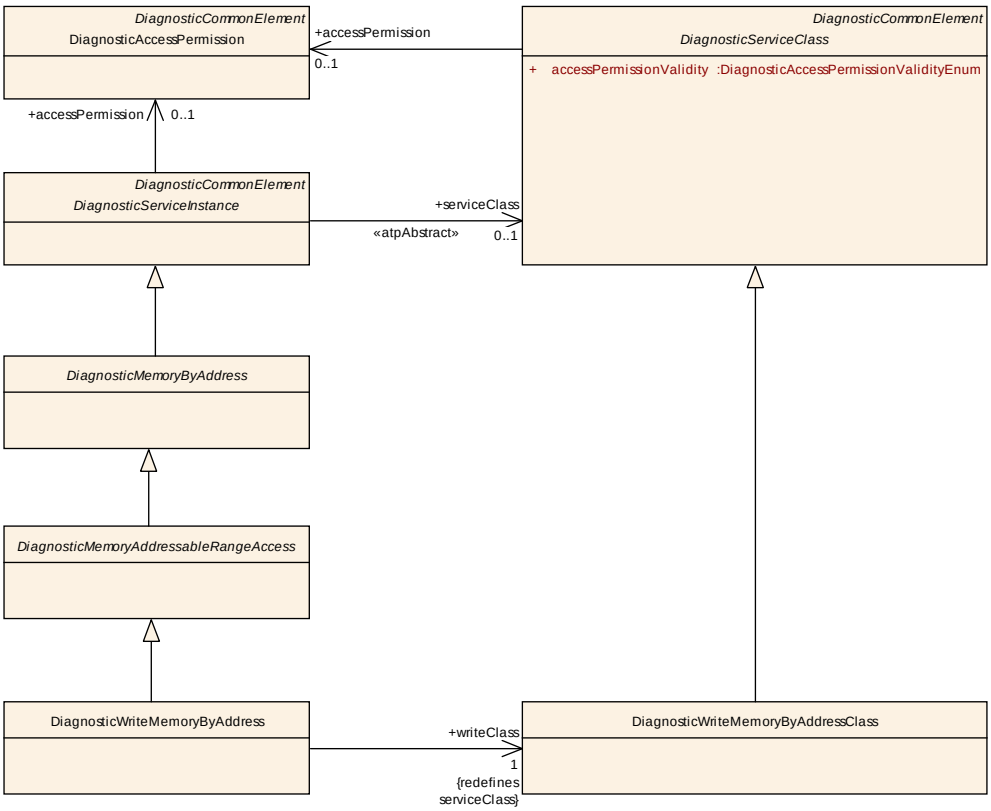

Figure 5.8: Modeling of diagnostic service `WriteMemoryByAddress` (0x3D)

[TPS_DEXT_01023] `WriteMemoryByAddress` does not define any sub-functions
⌈ The diagnostic service `WriteMemoryByAddress` does not define any sub-functions, therefore the value of `DiagnosticWriteMemoryByAddress.category` does not need to be constrained. ⌋*(RS_DEXT_00001, RS_DEXT_00020, RS_DEXT_00051)*

Class	**DiagnosticReadMemoryByAddress**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::MemoryByAddress			
Note	This represents an instance of the "Read Memory by Address" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticMemoryAddressableRangeAccess,DiagnosticMemoryByAddress,DiagnosticServiceInstance,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
readClass	DiagnosticReadMemoryByAddressClass	1	ref	This reference substantiates that abstract reference in the role serviceClass for this specific concrete class. Thereby, the reference represents the ability to access shared attributes among all DiagnosticReadMemoryByAddresst in the given context.

Table 5.26: DiagnosticReadMemoryByAddress

Class	**DiagnosticReadMemoryByAddressClass**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::MemoryByAddress			
Note	This meta-class contains attributes shared by all instances of the "Read Memory by Address" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticServiceClass,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
–	–	–	–	–

Table 5.27: DiagnosticReadMemoryByAddressClass

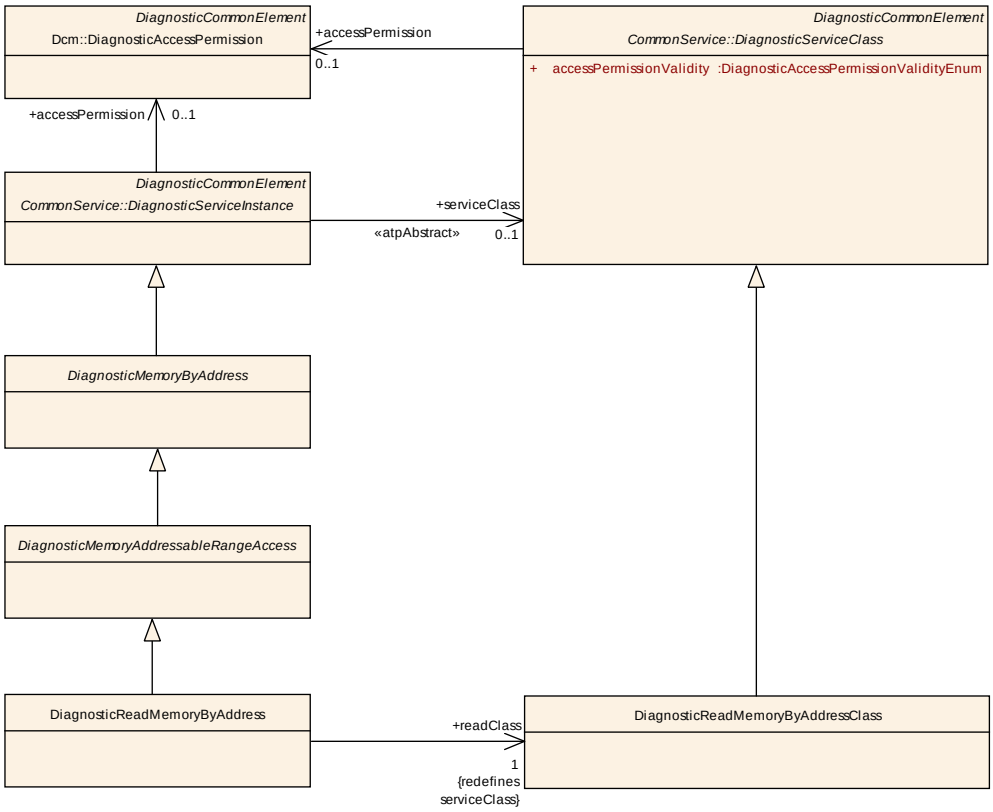

Figure 5.9: Modeling of diagnostic service `ReadMemoryByAddress` (0x23)

[TPS_DEXT_01024] `ReadMemoryByAddress` does not define any sub-functions ⌈ The diagnostic service `ReadMemoryByAddress` does not define any sub-functions, therefore the value of `DiagnosticReadMemoryByAddress.category` does not need to be constrained. ⌋*(RS_DEXT_00001, RS_DEXT_00008, RS_DEXT_00051)*

Class	**DiagnosticTransferExit**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::MemoryByAddress			
Note	This represents an instance of the "Transfer Exit" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticMemoryByAddress,DiagnosticServiceInstance,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
transferExitClass	DiagnosticTransferExitClass	1	ref	This reference substantiates that abstract reference in the role serviceClass for this specific concrete class. Thereby, the reference represents the ability to access shared attributes among all DiagnosticTransferExit in the given context.

Table 5.28: DiagnosticTransferExit

Class	**DiagnosticTransferExitClass**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::MemoryByAddress			
Note	This meta-class contains attributes shared by all instances of the "Transfer Exit" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticServiceClass,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
–	–	–	–	–

Table 5.29: DiagnosticTransferExitClass

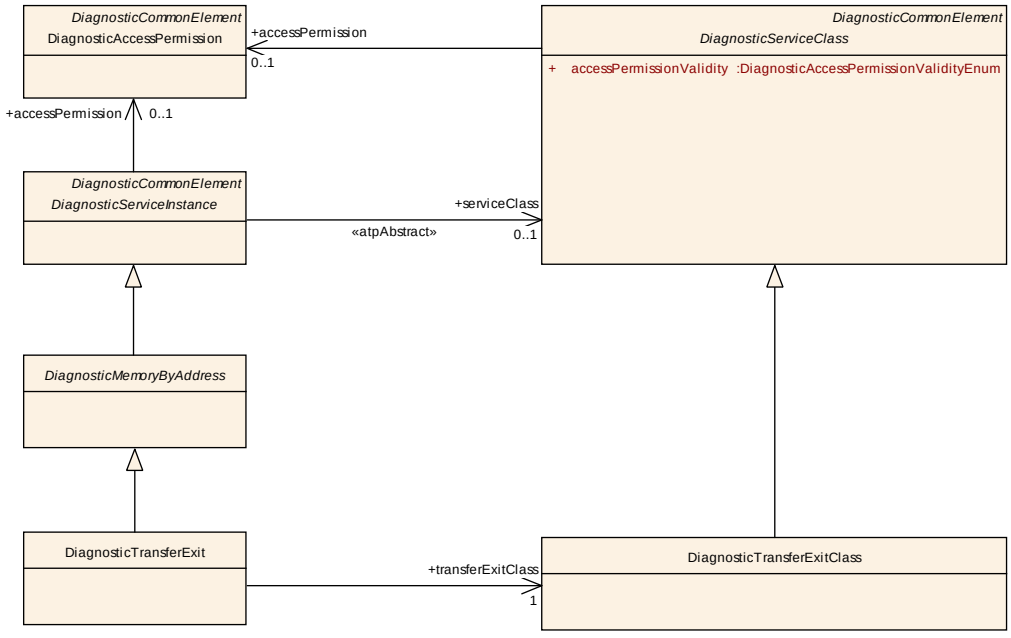

Figure 5.10: Modeling of diagnostic service `TransferExit` (0x37)

[TPS_DEXT_01025] `TransferExit` does not define any sub-functions ⌈ The diagnostic service `TransferExit` does not define any sub-functions, therefore the value of `DiagnosticTransferExit.category` does not need to be constrained. ⌋*(RS_DEXT_00001, RS_DEXT_00019, RS_DEXT_00051)*

Class	**DiagnosticDataTransfer**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::MemoryByAddress			
Note	This represents an instance of the "Data Transfer" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticMemoryByAddress,DiagnosticServiceInstance,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
dataTransferClass	DiagnosticDataTransferClass	1	ref	This reference substantiates that abstract reference in the role serviceClass for this specific concrete class. Thereby, the reference represents the ability to access shared attributes among all DiagnosticDataTransfer in the given context.

Table 5.30: DiagnosticDataTransfer

Class	**DiagnosticDataTransferClass**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::MemoryByAddress			
Note	This meta-class contains attributes shared by all instances of the "Data Transfer" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticServiceClass,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
–	–	–	–	–

Table 5.31: DiagnosticDataTransferClass

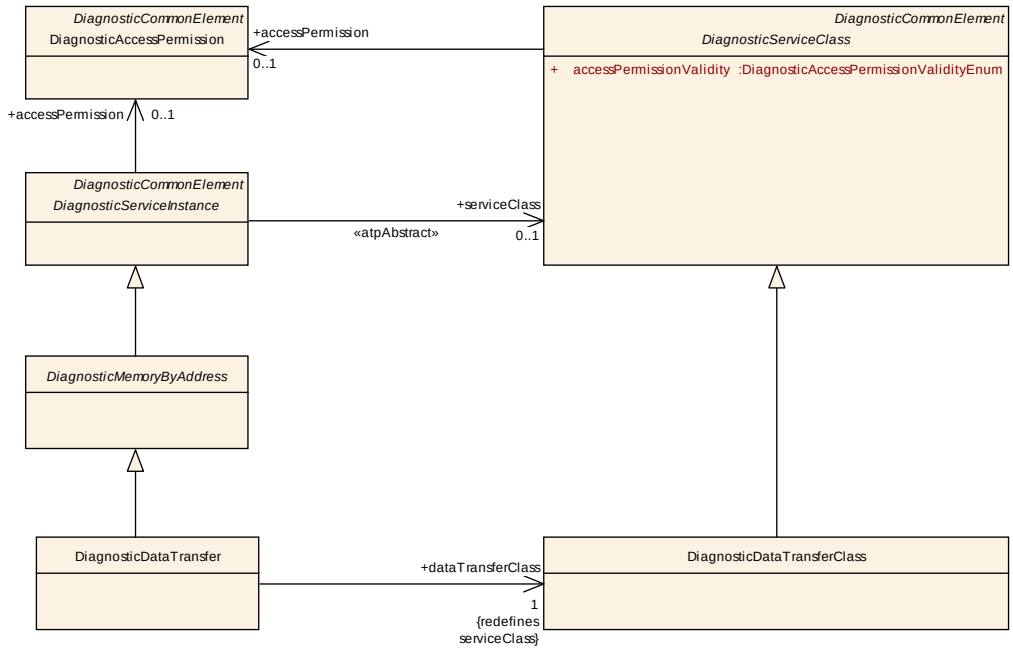

Figure 5.11: Modeling of diagnostic service `DataTransfer` (0x36)

[TPS_DEXT_01026] `DataTransfer` does not define any sub-functions ⌈ The diagnostic service `DataTransfer` does not define any sub-functions, therefore the value of `DiagnosticDataTransfer.category` does not need to be constrained. ⌋*(RS_DEXT_00001, RS_DEXT_00018, RS_DEXT_00051)*

Class	**DiagnosticRequestDownload**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::MemoryByAddress			
Note	This represents an instance of the "Request Download" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticMemoryAddressableRangeAccess,DiagnosticMemoryByAddress,DiagnosticServiceInstance,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
requestDownloadClass	DiagnosticRequestDownloadClass	1	ref	This reference substantiates that abstract reference in the role serviceClass for this specific concrete class. Thereby, the reference represents the ability to access shared attributes among all DiagnosticRequestDownload in the given context.

Table 5.32: DiagnosticRequestDownload

Class	**DiagnosticRequestDownloadClass**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::MemoryBy Address			
Note	This meta-class contains attributes shared by all instances of the "Request Download" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Diagnostic ServiceClass,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
–	–	–	–	–

Table 5.33: DiagnosticRequestDownloadClass

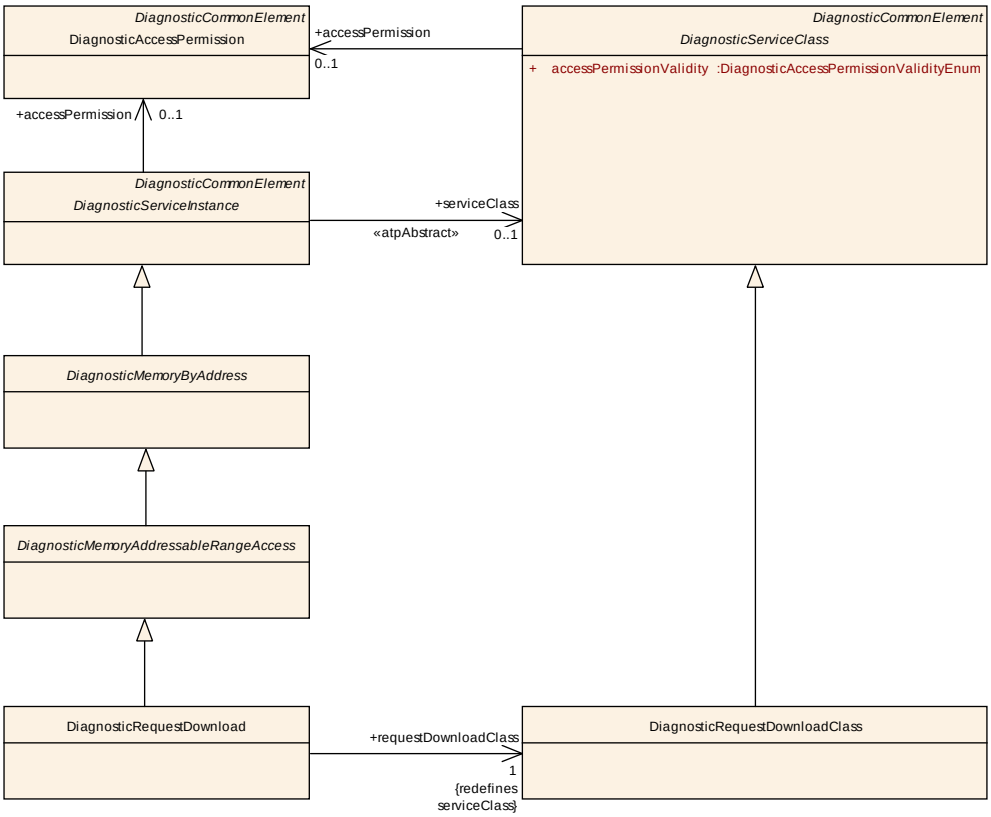

Figure 5.12: Modeling of diagnostic service `RequestDownload` (0x34)

[TPS_DEXT_01027] `RequestDownload` does not define any sub-functions ⌈ The diagnostic service `RequestDownload` does not define any sub-functions, therefore the value of `DiagnosticRequestDownload.category` does not need to be constrained. ⌋*(RS_DEXT_00001, RS_DEXT_00016, RS_DEXT_00051)*

Class	**DiagnosticRequestUpload**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::MemoryByAddress			
Note	This represents an instance of the "Request Upload" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticMemoryAddressableRangeAccess,DiagnosticMemoryByAddress,DiagnosticServiceInstance,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
requestUploadClass	DiagnosticRequestUploadClass	1	ref	This reference substantiates that abstract reference in the role serviceClass for this specific concrete class. Thereby, the reference represents the ability to access shared attributes among all DiagnosticRequestUpload in the given context.

Table 5.34: DiagnosticRequestUpload

Class	**DiagnosticRequestUploadClass**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::MemoryByAddress			
Note	This meta-class contains attributes shared by all instances of the "Request Upload" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticServiceClass,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
–	–	–	–	–

Table 5.35: DiagnosticRequestUploadClass

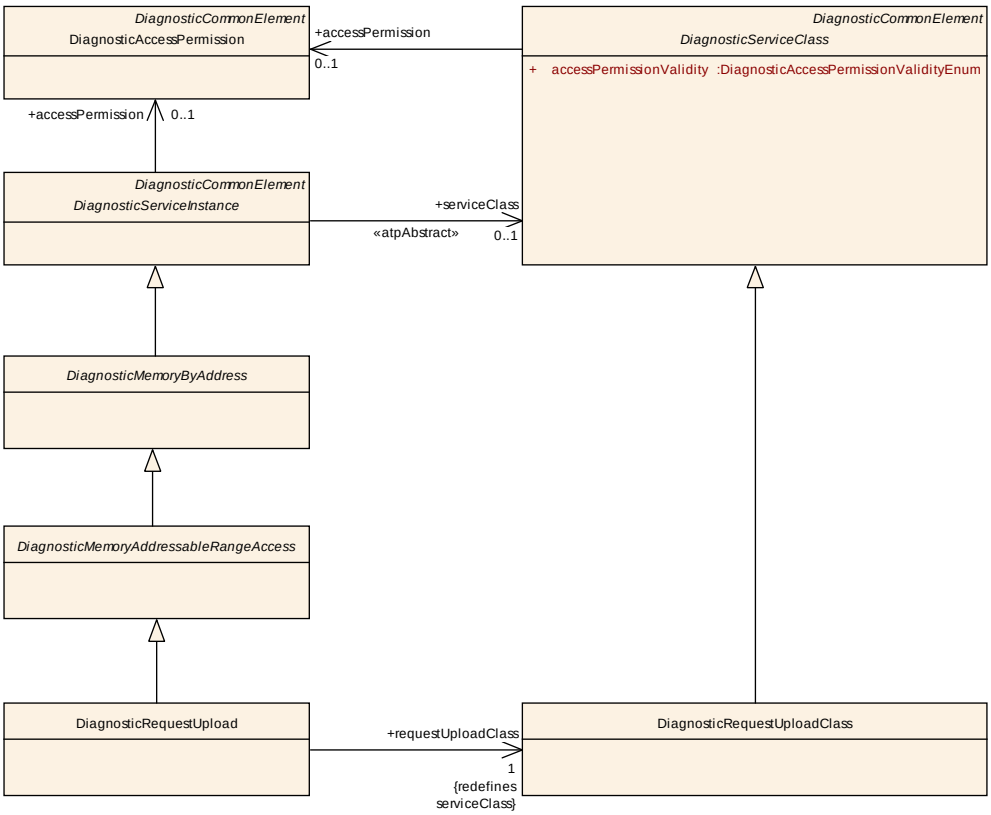

Figure 5.13: Modeling of diagnostic service `RequestDownload` (0x35)

[TPS_DEXT_01028] `RequestUpload` does not define any sub-functions ⌈ The diagnostic service `RequestUpload` does not define any sub-functions, therefore the value of `DiagnosticRequestUpload.category` does not need to be constrained. ⌋*(RS_DEXT_00001, RS_DEXT_00017, RS_DEXT_00051)*

5.4.6 CommunicationControl

This chapter describes the modeling of diagnostic services `CommunicationControl` (0x28). The purpose of this diagnostic service is to enable or disable `ISignalIPduGroup`s.

However, the actual implementation of the enabling or disabling is obviously not executed directly within the diagnostic stack. It requires some interaction with the BswM that in turn implements the enabling algorithm.

Therefore, the meta-class modeled for this purpose does not need to refer to `ISignalIPduGroup`s but implements a mode request to the BswM.

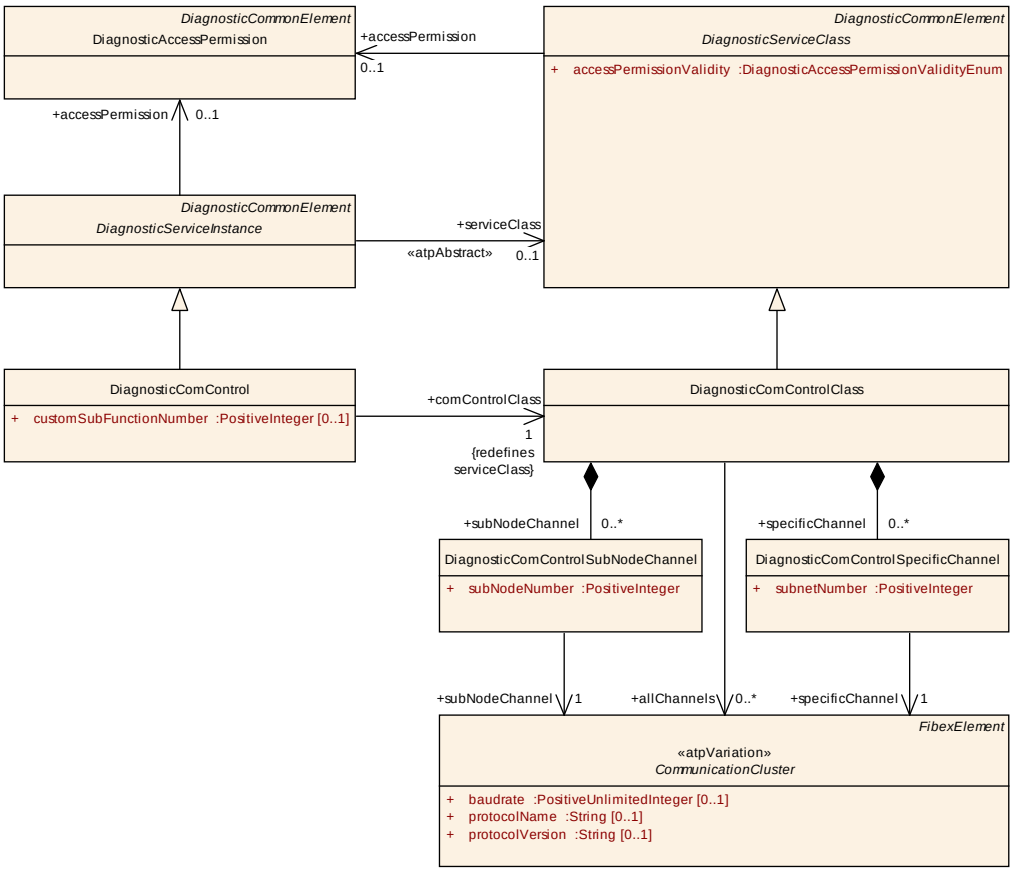

Figure 5.14: Modeling of diagnostic service `CommunicationControl` (0x28)

[TPS_DEXT_01057] Allowed values of `DiagnosticComControl.category` ⌈ The sub-functions of the diagnostic services `CommunicationControl` are identified by means of the attribute `DiagnosticComControl.category`. Standardized values of `DiagnosticComControl.category` are:

- `ENABLE_RX_AND_TX`
- `DISABLE_RX_AND_TX`
- `ENABLE_RX_AND_DISABLE_TX`
- `DISABLE_RX_AND_ENABLE_TX`
- `ENABLE_RX_AND_DISABLE_TX_WITH_ENHANCED_ADDRESS_INFORMATION`
- `ENABLE_RX_AND_TX_WITH_ENHANCED_ADDRESS_INFORMATION`

The meaning of these values is described in the applicable ISO document [15]. ⌋*(RS_DEXT_00001, RS_DEXT_00010, RS_DEXT_00051)*

[TPS_DEXT_01029] Correspondence of `category` values to numerical values mentioned in the ISO 14229-1 ⌈ The ISO 14229-1 [15] standard document defines specific numerical values for the sub-functions of the diagnostic service `CommunicationControl`.

The correspondence of the numerical values to the pre-defined values of category according to [TPS_DEXT_01057] is pretty obvious because the definition of values defined in [TPS_DEXT_01057] has been directly inspired by the ISO 14229-1 [15] standard document. ⌋*(RS_DEXT_00001, RS_DEXT_00010, RS_DEXT_00051)*

[TPS_DEXT_01030] Manufacturer-specific values for sub-functions of service CommunicationControl ⌈ The ISO 14229-1 [15] standard document, beyond the standardized numerical values for sub-functions, reserves a numerical range of sub-Function identifiers for manufacturer-specific use.

In this case it is possible to define further values for category, provided that a custom prefix is used to avoid potential name clashes with further extensions of the AUTOSAR standard, namely [TPS_DEXT_01057]. ⌋*(RS_DEXT_00010, RS_DEXT_00047, RS_DEXT_00051)*

[TPS_DEXT_01031] Semantics of DiagnosticComControl.customSubFunctionNumber ⌈ The attribute DiagnosticComControl.customSubFunctionNumber has been introduced to allow for the specification of a manufacturer-or supplier-specific value to represent the custom sub-function in the diagnostic communication.

The tuple created by the values of attributes DiagnosticComControl.category and DiagnosticComControl.customSubFunctionNumber fully specifies identification of the manufacturer- or supplier-specific sub-function. ⌋*(RS_DEXT_00010, RS_DEXT_00047, RS_DEXT_00051)*

[constr_1334] Existence of DiagnosticComControl.customSubFunctionNumber ⌈ The attribute DiagnosticComControl.customSubFunctionNumber shall only exist if the value of DiagnosticComControl.category is outside the standardized set of values as defined by [TPS_DEXT_01057]. ⌋*()*

[constr_1335] Possible values for DiagnosticComControl.customSubFunctionNumber ⌈ Given the fulfillment of [constr_1334], the value of a given DiagnosticComControl.customSubFunctionNumber shall always be within the closed interval **0x40** .. **0x5F** (for manufacturer-specific sub-functions) or the closed interval **0x60** .. **0x7E** (for supplier-specific sub-functions). ⌋*()*

[TPS_DEXT_01032] Impact of the DiagnosticComControlClass on the state management for CommunicationClusters ⌈ The impact of the DiagnosticComControlClass on the state management for CommunicationClusters can have two alternative consequences:

- All CommunicationClusters are affected. For this purpose the attribute allChannels has the ability to identify the applicable CommunicationClusters.

  It may seem counterintuitive to require a reference to all applicable CommunicationClusters when the expected semantics is actually to define an impact on **all** of them.

  However, there could be private CommunicationClusters that are not participating in the diagnostics work-flow: These need to be kept out of scope and there-

fore the explicit identification of applicable CommunicationClusters makes sense.

- A selected number of CommunicationClusters is affected. This is conceptually different from the other use case in that it requires an additional attribute that keeps a subnetNumber that is typically assigned by the OEM role.

⌋(*RS_DEXT_00010*)

[constr_1336] Applicable value range for DiagnosticComControlSpecificChannel.subnetNumber ⌈ The value of attribute DiagnosticComControlSpecificChannel.subnetNumber shall be within the closed interval **1** .. **14**. ⌋()

Please note that the regulation implied by [constr_1336] has not been introduced on an arbitrary basis but gets its conceptual background from the ISO 14229-1 [15] standard document.

Obviously, a diagnostic service with the name CommunicationControl will have an impact on the enclosing ECU's mode management. This impact, however, is not defined by any further attributes or references, the DiagnosticComControl **is** the impact.

By defining a DiagnosticComControl and setting the category to one of the applicable values (e.g. ENABLE_RX_AND_TX), it is possible to express the intended semantics to the full extent.

[constr_1337] Allowed value range for attribute DiagnosticComControlSubNodeChannel.subNodeNumber ⌈ The value of attribute DiagnosticComControlSubNodeChannel.subNodeNumber shall not exceed the closed interval **0** .. **65535**. ⌋()

[TPS_DEXT_01074] Difference between the attributes DiagnosticComControl.specificChannel and DiagnosticComControl.subNodeChannel ⌈ The semantical difference between the attributes DiagnosticComControl.specificChannel and DiagnosticComControl.subNodeChannel is that DiagnosticComControl.specificChannel actually refers to a CommunicationCluster whereas DiagnosticComControl.subNodeChannel basically refers to a CommunicationCluster to which the nodes with the given identification numbers are connected. ⌋(*RS_DEXT_00010*)

Class	**DiagnosticComControl**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::CommunicationControl			
Note	This represents an instance of the "Communication Control" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticServiceInstance,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***

Attribute	Datatype	Mul.	Kind	Note
comControlClass	DiagnosticComControlClass	1	ref	This reference substantiates that abstract reference in the role serviceClass for this specific concrete class. Thereby, the reference represents the ability to access shared attributes among all DiagnosticComControl in the given context.
customSubFunctionNumber	PositiveInteger	0..1	attr	This attribute shall be used to define a custom sub-function number if none of the standardized values of category shall be used.

Table 5.36: DiagnosticComControl

Class	DiagnosticComControlSpecificChannel			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::CommunicationControl			
Note	This represents the ability to add further attributes to the definition of a specific channel that is subject to the diagnostic service "communication control".			
Base	ARObject			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
specificChannel	CommunicationCluster	1	ref	This represents the affected CommunicationClusters in the role specificChannel
subnetNumber	PositiveInteger	1	attr	This represents the applicable subnet number (which is an arbitrary number ranging from 1..14)

Table 5.37: DiagnosticComControlSpecificChannel

Class	DiagnosticComControlClass			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::CommunicationControl			
Note	This meta-class contains attributes shared by all instances of the "Communication Control" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticServiceClass,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
allChannels	CommunicationCluster	*	ref	This reference represents the semantics that all available channels shall be affected. It is still necessary to refer to individual CommunicatuionClusters because there could be private CommunicationClusters in the System Extract that are not subject to the service "communication control". By referring to the applicable CommunicationClusters it can be made sure that only the affected CommunicationClusters are accessed.

Attribute	Datatype	Mul.	Kind	Note
specificChannel	DiagnosticComControlSpecificChannel	*	aggr	This represents the ability to add additional attributes to the case that only specific channels are supposed to be considered,
subNodeChannel	DiagnosticComControlSubNodeChannel	*	aggr	

Table 5.38: DiagnosticComControlClass

Class	DiagnosticComControlSubNodeChannel			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::CommunicationControl			
Note	This represents the ability to add further attributes to the definition of a specific sub-node channel that is subject to the diagnostic service "communication control".			
Base	ARObject			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
subNodeChannel	CommunicationCluster	1	ref	This represents the affected CommunicationClusters in the role subNodeChannel
subNodeNumber	PositiveInteger	1	attr	This represents the applicable subNode number. The value corresponds to the request message parameter nodeIdentificationNumber of diagnostic service CommunicationControl (0x28).

Table 5.39: DiagnosticComControlSubNodeChannel

5.4.7 DynamicallyDefineDataIdentifier

This chapter describes the modeling of diagnostic services `DynamicallyDefineDataIdentifier` (0x2C). The purpose of the service is to allow for defining data identifiers (DID) at run-time.

By this means it is possible to combine existing diagnostic data into a single DID.

This semantics is reflected by the modeling of the meta-class `DiagnosticDynamicallyDefineDataIdentifier` that refers to a `DiagnosticDynamicDataIdentifier` in the role `dataIdentifier`.

Also, the `DiagnosticDynamicallyDefineDataIdentifier` inherits a reference to `accessPermission` from `DiagnosticServiceInstance`.

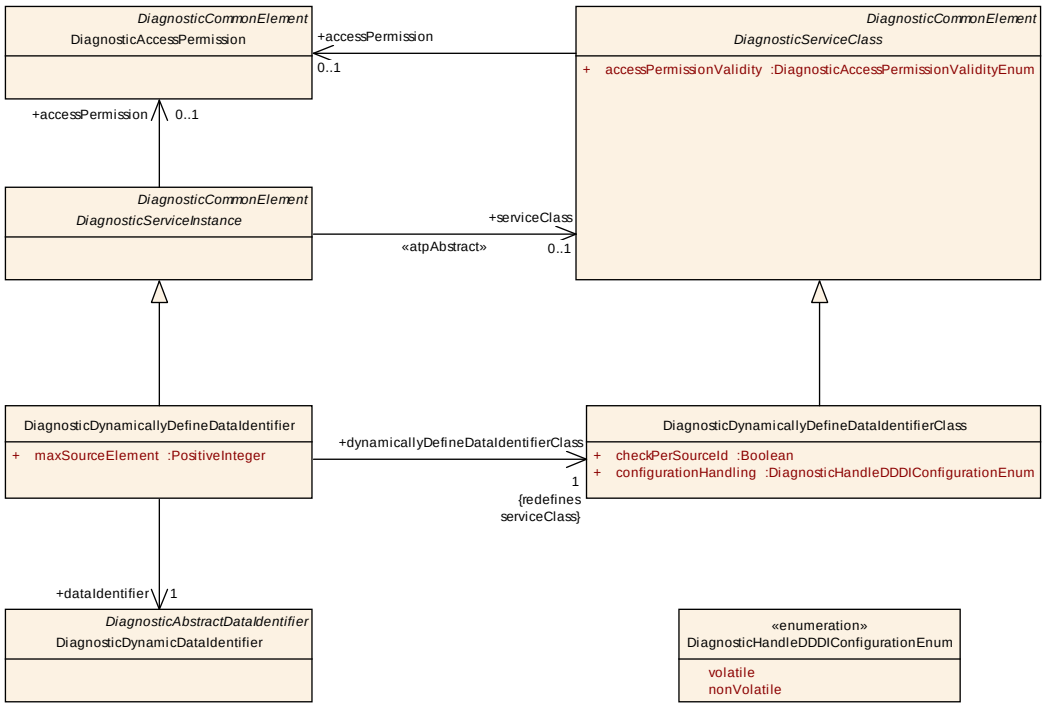

Figure 5.15: Modeling of diagnostic service `DynamicallyDefineDataIdentifier` (0x2C)

[TPS_DEXT_01058] Standardized values for `DiagnosticDynamicallyDefineDataIdentifier.category` ⌈ The following values of the attribute `DiagnosticDynamicallyDefineDataIdentifier.category` are standardized by AUTOSAR:

- `DEFINE_BY_IDENTIFIER`
- `DEFINE_BY_MEMORY_ADDRESS`
- `CLEAR_DYNAMICALLY_DEFINED_DATA_IDENTIFIER`

The meaning of these values is described in the applicable ISO document [15]. ⌋*(RS_DEXT_00001, RS_DEXT_00012, RS_DEXT_00051)*

Class	**DiagnosticDynamicallyDefineDataIdentifier**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::DynamicallyDefineDataIdentifier			
Note	This represents an instance of the "Dynamically Define Data Identifier" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticServiceInstance,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
dataIdentifier	DiagnosticDynamicDataIdentifier	1	ref	This represents the applicable DiagnosticDynamicDataIdentfier.

Attribute	Datatype	Mul.	Kind	Note
dynamicallyDefineDataIdentifierClass	DiagnosticDynamicallyDefineDataIdentifierClass	1	ref	This reference substantiates that abstract reference in the role serviceClass for this specific concrete class. Thereby, the reference represents the ability to access shared attributes among all DiagnosticDynamicallyDefineDataIdentifier in the given context.
maxSourceElement	PositiveInteger	1	attr	This represents the maximum number of source elements of the dynamically created DID.

Table 5.40: DiagnosticDynamicallyDefineDataIdentifier

Class	DiagnosticDynamicallyDefineDataIdentifierClass			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::DynamicallyDefineDataIdentifier			
Note	This meta-class contains attributes shared by all instances of the "Dynamically Define Data Identifier" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticServiceClass,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
checkPerSourceId	Boolean	1	attr	If set to TRUE, the Dcm module shall check the session, security and mode dependencies per source DIDs with a ReadDataByIdentifier (0x22) with DID in the range 0xF200 to 0xF3FF. If set to FALSE. the Dcm module shall not check the session, security and mode dependencies per source DIDs with a ReadDataByIdentifier (0x22) with DID in the range 0xF200 to 0xF3FF.
configurationHandling	DiagnosticHandleDDDIConfigurationEnum	1	attr	This configuration switch defines whether DDDID definition is handled as non-volatile information or not.

Table 5.41: DiagnosticDynamicallyDefineDataIdentifierClass

5.4.8 ReadDataByPeriodicIdentifier

This chapter describes the modeling of diagnostic services `ReadDataByPeriodicIdentifier` (0x2A).

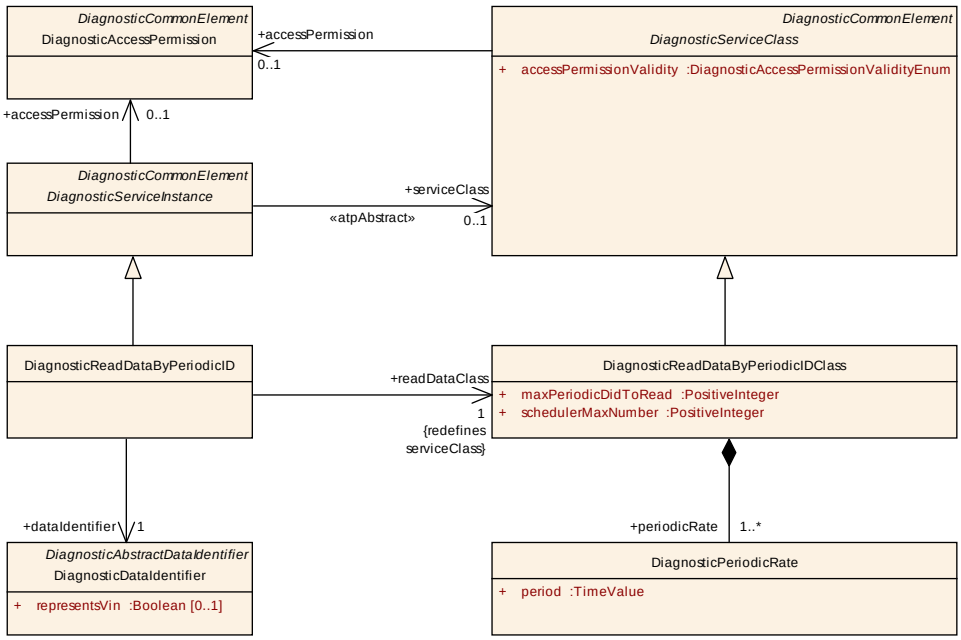

Figure 5.16: Modeling of diagnostic service `ReadDataByPeriodicIdentifier` (0x2A)

[TPS_DEXT_01059] Applicable values for `DiagnosticPeriodicRate.category`
⌈ The following values of the attribute `DiagnosticPeriodicRate.category` are standardized by AUTOSAR:

- `PERIODIC_RATE_FAST`
- `PERIODIC_RATE_MEDIUM`
- `PERIODIC_RATE_SLOW`

The meanings of these values are described in the applicable ISO document [15]. ⌋*(RS_DEXT_00001, RS_DEXT_00011, RS_DEXT_00051)*

[constr_1338] Maximum number of aggregated `DiagnosticReadDataByPeriodicIDClass.periodicRate` ⌈ The number of aggregated `periodicRate` within the context of one `DiagnosticReadDataByPeriodicIDClass` shall be within the closed interval 1..3. ⌋*()*

Class	**DiagnosticReadDataByPeriodicID**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::ReadDataByPeriodicID			
Note	This represents an instance of the "Read Data by periodic Identifier" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticServiceInstance,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
dataIdentifier	DiagnosticDataIdentifier	1	ref	This represents the corresponding DiagnosticDataIdentifier.

Attribute	Datatype	Mul.	Kind	Note
readDataClass	DiagnosticReadDataByPeriodicIDClass	1	ref	This reference substantiates that abstract reference in the role serviceClass for this specific concrete class. Thereby, the the reference represents the ability to access shared attributes among all DiagnosticReadDataByPeriodicID in the given context.

Table 5.42: DiagnosticReadDataByPeriodicID

Class	DiagnosticReadDataByPeriodicIDClass			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::ReadDataByPeriodicID			
Note	This meta-class contains attributes shared by all instances of the "Read Data by periodic Identifier" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticServiceClass,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
maxPeriodicDidToRead	PositiveInteger	1	attr	This represents the maximum number of data identifiers that can be included in one request.
periodicRate	DiagnosticPeriodicRate	1..*	aggr	This represents the description of a collection of periodic rates in which the service can be executed.
schedulerMaxNumber	PositiveInteger	1	attr	This represents the maximum number of periodic data identifiers that can be scheduled in parallel.

Table 5.43: DiagnosticReadDataByPeriodicIDClass

Class	DiagnosticPeriodicRate			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::ReadDataByPeriodicID			
Note	This represents the ability to define a periodic rate for the specification of the "read data by periodic ID" diagnostic service.			
Base	ARObject			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
period	TimeValue	1	attr	This represents the period of the DiagnosticPeriodicRate in seconds.

Table 5.44: DiagnosticPeriodicRate

5.4.9 ControlDTCSetting

This chapter describes the modeling of diagnostic services `ControlDTCSetting` (0x85). The purpose of the diagnostic service is to let the tester tell the diagnostic stack to either stop or resume the updating of a diagnostic trouble code.

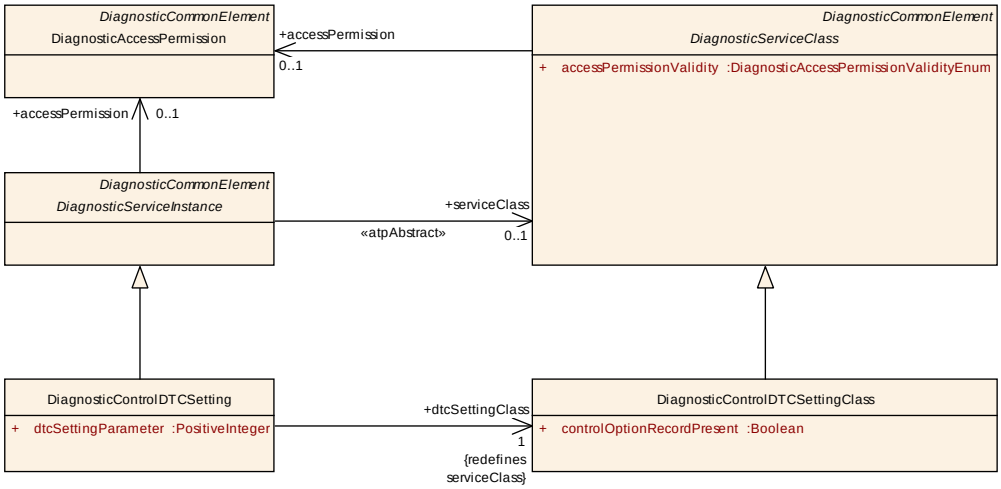

Figure 5.17: Modeling of diagnostic service `ControlDTCSetting` (0x85)

[TPS_DEXT_01075] standardized values for the attribute `DiagnosticControlDTCSetting.category` ⌈ AUTOSAR does not standardize any of the possible values for the attribute `DiagnosticControlDTCSetting.category`. ⌋*(RS_DEXT_00001, RS_DEXT_00021, RS_DEXT_00051)*

[TPS_DEXT_01076] Identification of sub-functions of diagnostic service `ControlDTCSetting` ⌈ The identification of sub-functions (for which the attribute `category` is used for several other `DiagnosticServiceInstance`s) is done via the attribute `DiagnosticControlDTCSetting.dtcSettingParameter`.

⌋*(RS_DEXT_00021, RS_DEXT_00051)*

Class	**DiagnosticControlDTCSetting**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::ControlDTCSetting			
Note	This represents an instance of the "Control DTC Setting" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticServiceInstance,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
dtcSettingClass	DiagnosticControlDTCSettingClass	1	ref	This reference substantiates that abstract reference in the role serviceClass for this specific concrete class. Thereby, the the reference represents the ability to access shared attributes among all DiagnosticControlDTCSetting in the given context.

Attribute	Datatype	Mul.	Kind	Note
dtcSetting Parameter	PositiveInteger	1	attr	This represents the DTCSettingType defined by ISO 14229-1. The pre-defined values are 1 (ON) and 2 (OFF).

Table 5.45: DiagnosticControlDTCSetting

Class	DiagnosticControlDTCSettingClass			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::ControlDTC Setting			
Note	This meta-class contains attributes shared by all instances of the "Control DTC Setting" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Diagnostic ServiceClass,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
controlOpti onRecordP resent	Boolean	1	attr	This represents the decision whether the DTCSettingControlOptionRecord (see ISO 14229-1) is in general supported in the request message.

Table 5.46: DiagnosticControlDTCSettingClass

5.4.10 ResponseOnEvent

This chapter describes the modeling of diagnostic services `ResponseOnEvent` (0x86). The purpose of this service is to instruct the AUTOSAR diagnostic stack with respect to the starting or stopping of sending responses to a specific event to the tester.

Each `DiagnosticResponseOnEvent` provides the ability to define a collection of triggers (modelled by means of the abstract meta-class `DiagnosticResponseOnEventTrigger`) that cause the sending of a response message.

The actual trigger behavior is defined by the sub-class of `DiagnosticResponseOnEventTrigger` used to specify whether the trigger shall be created in response to a data change or in response to a DTC change.

[TPS_DEXT_01033] Semantics of triggers in the context of a `DiagnosticResponseOnEvent` ⌈ The semantics of a trigger in the context of a `DiagnosticResponseOnEvent` can be defined in two ways:

- The meta-class `DiagnosticDataChangeTrigger` allows for defining a trigger that activates on the change of the value of the referenced (in the role `dataIdentifier`) `DiagnosticDataIdentifier`.
- The meta-class `DiagnosticDtcChangeTrigger` allows for defining a trigger for the activation of the service. The entire proceedings of how the trigger activates and what DTCs are affected in managed at run-time and therefore no further configuration is required at this point.

⌋(*RS_DEXT_00022*)

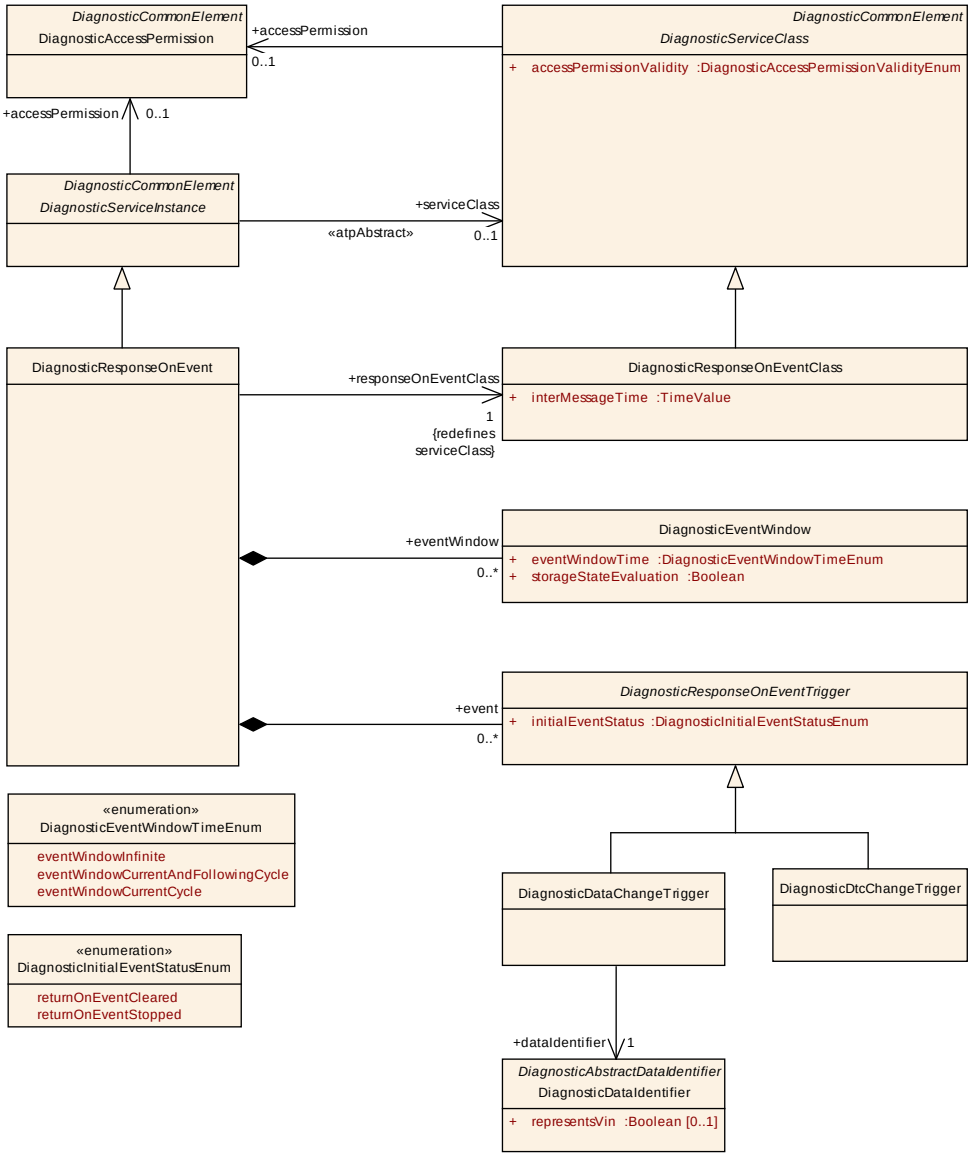

Figure 5.18: Modeling of diagnostic service `ResponseOnEvent` (0x86)

[constr_1365] Multiplicity of `DiagnosticResponseOnEvent.event` ⌈ The multiplicity of `DiagnosticResponseOnEvent.event` shall not exceed the upper bound **255**. ⌋*()*

[constr_1366] Event ID in the context of diagnostic service `ResponseOnEvent` shall be unique ⌈ The value of `DiagnosticResponseOnEvent.event.dataIdentifier.id` shall be unique within the context of a given `DiagnosticResponseOnEvent`. ⌋*()*

Class	**DiagnosticResponseOnEvent**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::ResponseOnEvent			
Note	This represents an instance of the "Response on Event" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticServiceInstance,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
event	DiagnosticResponseOnEventTrigger	*	aggr	This represents the collection of DiagnosticResponseOnEventTriggers defined in the context of the enclosing DiagnosticResponseOnEvent.
eventWindow	DiagnosticEventWindow	*	aggr	This represents the applicable DiagnosticEventWindows
responseOnEventClass	DiagnosticResponseOnEventClass	1	ref	This reference substantiates that abstract reference in the role serviceClass for this specific concrete class. Thereby, the reference represents the ability to access shared attributes among all DiagnosticResponseOnEvent in the given context.

Table 5.47: DiagnosticResponseOnEvent

Class	**DiagnosticResponseOnEventClass**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::ResponseOnEvent			
Note	This represents the ability to define common properties for alle instances of the "Response on Event" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticServiceClass,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
interMessageTime	TimeValue	1	attr	Provide the minimum time in seconds between two consecutive transmissions of an ROE event.

Table 5.48: DiagnosticResponseOnEventClass

Class	**DiagnosticEventWindow**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::ResponseOnEvent			
Note	This represents the ability to define the characteristics of the applicable event window			
Base	ARObject			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
eventWindowTime	DiagnosticEventWindowTimeEnum	1	attr	This attribute clarifies the validity of the eventWindow
storageStateEvaluation	Boolean	1	attr	If this attribute is set to TRUE the StorageStateBit will be evaluated if this EventWindowTime is requested.

Table 5.49: DiagnosticEventWindow

Class	**DiagnosticResponseOnEventTrigger (abstract)**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::ResponseOnEvent			
Note	This represents the ability to further specify the events that are associated with the execution of the diagnostic service.			
Base	ARObject			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
initialEventStatus	DiagnosticInitialEventStatusEnum	1	attr	This represents the initial status of the enclosing DiagnosticResponseOnEventTrigger.

Table 5.50: DiagnosticResponseOnEventTrigger

Class	**DiagnosticDataChangeTrigger**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::ResponseOnEvent			
Note	This represents the ability to define a trigger based on the change of a given DiagnosticDataIdentifier.			
Base	ARObject,DiagnosticResponseOnEventTrigger			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
dataIdentifier	DiagnosticDataIdentifier	1	ref	This represents the corresponding DiagnosticDataIdentifier.

Table 5.51: DiagnosticDataChangeTrigger

Class	**DiagnosticDtcChangeTrigger**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::ResponseOnEvent			
Note	This represents the ability to define a trigger that executes on the change of any DiagnosticTroubleCode.			
Base	ARObject,DiagnosticResponseOnEventTrigger			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
–	–	–	–	–

Table 5.52: DiagnosticDtcChangeTrigger

Enumeration	**DiagnosticInitialEventStatusEnum**
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::ResponseOnEvent
Note	This represents the ability to define an initial status for the ROE service.
Literal	***Description***
returnOnEventCleared	This means that the ResponseOnEvent is initially cleared.
returnOnEventStopped	This means that the ResponseOnEvent is initially stopped.

Table 5.53: DiagnosticInitialEventStatusEnum

Enumeration	**DiagnosticEventWindowTimeEnum**
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::ResponseOnEvent
Note	This represents the ability to define the semantics of the event window.
Literal	***Description***
eventWindowCurrentAndFollowingCycle	This means that the window extends to this and the following cycle.
eventWindowCurrentCycle	This means that the window is limited to the current cycle.
eventWindowInfinite	This means that the window extents without a border.

Table 5.54: DiagnosticEventWindowTimeEnum

5.4.11 ReadDTCInformation

This chapter describes the modeling of diagnostic services `ReadDTCInformation` (0x19). The purpose of this service is enable a tester to read a `Diagnostic Trouble Code` from the AUTOSAR `Dcm` [11] (that, in turn, fetches the information from the AUTOSAR `Dem` [12]).

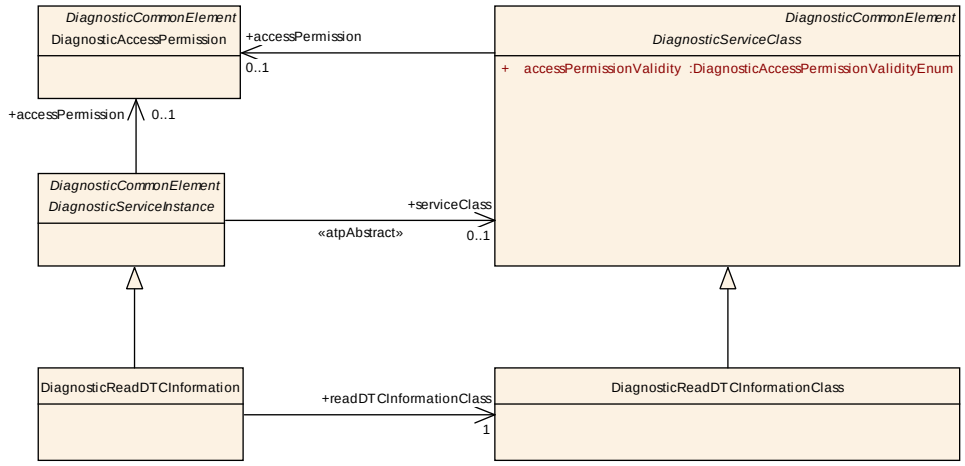

Figure 5.19: Modeling of diagnostic service `ReadDTCInformation` (0x19)

[TPS_DEXT_01034] Sub-functions of the service `ReadDTCInformation` ⌈ The diagnostics service `ReadDTCInformation` defines a number of sub-functions that are, as far as the modeling in AUTOSAR goes, identified by a textual identifier.

These sub-functions are **not** modeled explicitly but can be specified by defining a `DiagnosticReadDTCInformation` and by setting the `DiagnosticReadDTCInformation.category` to the identifier of the respective sub-function.

The possible values, as far as the AUTOSAR standard is concerned, are defined by [TPS_DEXT_01060]. ⌋*(RS_DEXT_00006, RS_DEXT_00051)*

[TPS_DEXT_01060] Applicable values for DiagnosticReadDTCInformation.category ⌈ The following values of the attribute DiagnosticReadDTCInformation.category are standardized by AUTOSAR:

- REPORT_NUMBER_OF_DTC_BY_STATUS_MASK
- REPORT_DTC_BY_STATUS_MASK
- REPORT_MIRROR_MEMORY_DTC_BY_STATUS_MASK
- REPORT_NUMBER_OF_MIRROR_MEMORY_DTC_BY_STATUS_MASK
- REPORT_NUMBER_OF_EMISSIONS_OBD_DTC_BY_STATUS_MASK
- REPORT_EMISSIONS_OBD_DTC_BY_STATUS_MASK
- REPORT_DTC_SNAPSHOT_IDENTIFICATION
- REPORT_DTC_SNAPSHOT_RECORD_BY_DTC_NUMBER
- REPORT_DTC_STORED_DATA_BY_RECORD_NUMBER
- REPORT_DTC_EXT_DATA_RECORD_BY_DTC_NUMBER
- REPORT_MIRROR_MEMORY_DTC_EXT_DATA_RECORD_BY_DTC_NUMBER
- REPORT_NUMBER_OF_DTC_BY_SEVERITY_MASK_RECORD
- REPORT_DTC_BY_SEVERITY_MASK_RECORD
- REPORT_SEVERITY_INFORMATION_OF_DTC
- REPORT_SUPPORTED_DTC
- REPORT_FIRST_TEST_FAILED_DTC
- REPORT_FIRST_CONFIRMED_DTC
- REPORT_MOST_RECENT_TEST_FAILED_DTC
- REPORT_MOST_RECENT_CONFIRMED_DTC
- REPORT_DTC_FAULT_DETECTION_COUNTER
- REPORT_DTC_WITH_PERMANENT_STATUS
- REPORT_USER_DEF_MEMORY_DTC_BY_STATUS_MASK
- REPORT_USER_DEF_MEMORY_DTC_SNAPSHOT_RECORD_BY_DTC_NUMBER
- REPORT_USER_DEF_MEMORY_DTC_EXT_DATA_RECORD_BY_DTC_NUMBER
- REPORT_WWH_OBD_DTC_BY_MASK_RECORD

The meanings of these values are described in the applicable ISO document (ISO 14229-1) [15]. ⌋*(RS_DEXT_00001, RS_DEXT_00006, RS_DEXT_00051)*

Please note that there is nothing to configure for DiagnosticReadDTCInformation beyond its mere existence.

Class	**DiagnosticReadDTCInformation**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::ReadDTC Information			
Note	This represents an instance of the "Read DTC Information" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Diagnostic ServiceInstance,Identifiable,MultilanguageReferrable,Packageable Element,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
readDTCIn formationC lass	DiagnosticRead DTCInformation Class	1	ref	This reference substantiates that abstract reference in the role serviceClass for this specific concrete class. Thereby, the reference represents the ability to access shared attributes among all DiagnosticReadDTCInformation in the given context.

Table 5.55: DiagnosticReadDTCInformation

5.4.12 RoutineControl

This chapter describes the modeling of diagnostic services RoutineControl (0x31). The purpose of this diagnostic service is to execute a piece of code, a Diagnostic Routine, on the diagnostic stack at the request of the tester.

Diagnostic Routines consist of up to three possible components:

- Start Routine
- Stop Routine
- Request Routine Results

The impact of this architecture no the meta-model is described by [TPS_DEXT_01077]:

[TPS_DEXT_01077] Modeling of DiagnosticRoutine ⌈ From the meta-modeling point of view, the semantics of DiagnosticRoutine is created by aggregating three further meta-classes:

- DiagnosticStartRoutine
- DiagnosticStopRoutine
- DiagnosticRequestRoutineResults

⌋(*RS_DEXT_00015*)

[TPS_DEXT_01088] Semantics of `DiagnosticRoutine.id` ⌈ The attribute `DiagnosticRoutine.id` represents the so-called *identifier* of the `DiagnosticRoutine`. ⌋*(RS_DEXT_00036)*

[TPS_DEXT_01078] Not possible to use the attribute `category` for the identification of the sub-function of diagnostic service `RoutineControl` ⌈ In the case of `DiagnosticRoutine`, it is not possible to use the attribute `category` for the identification of the sub-function. ⌋*(RS_DEXT_00015, RS_DEXT_00051)*

The sub-functions actually have different properties i.e. the arguments to a `DiagnosticRoutine`) that require a dedicated modeling for this purpose.

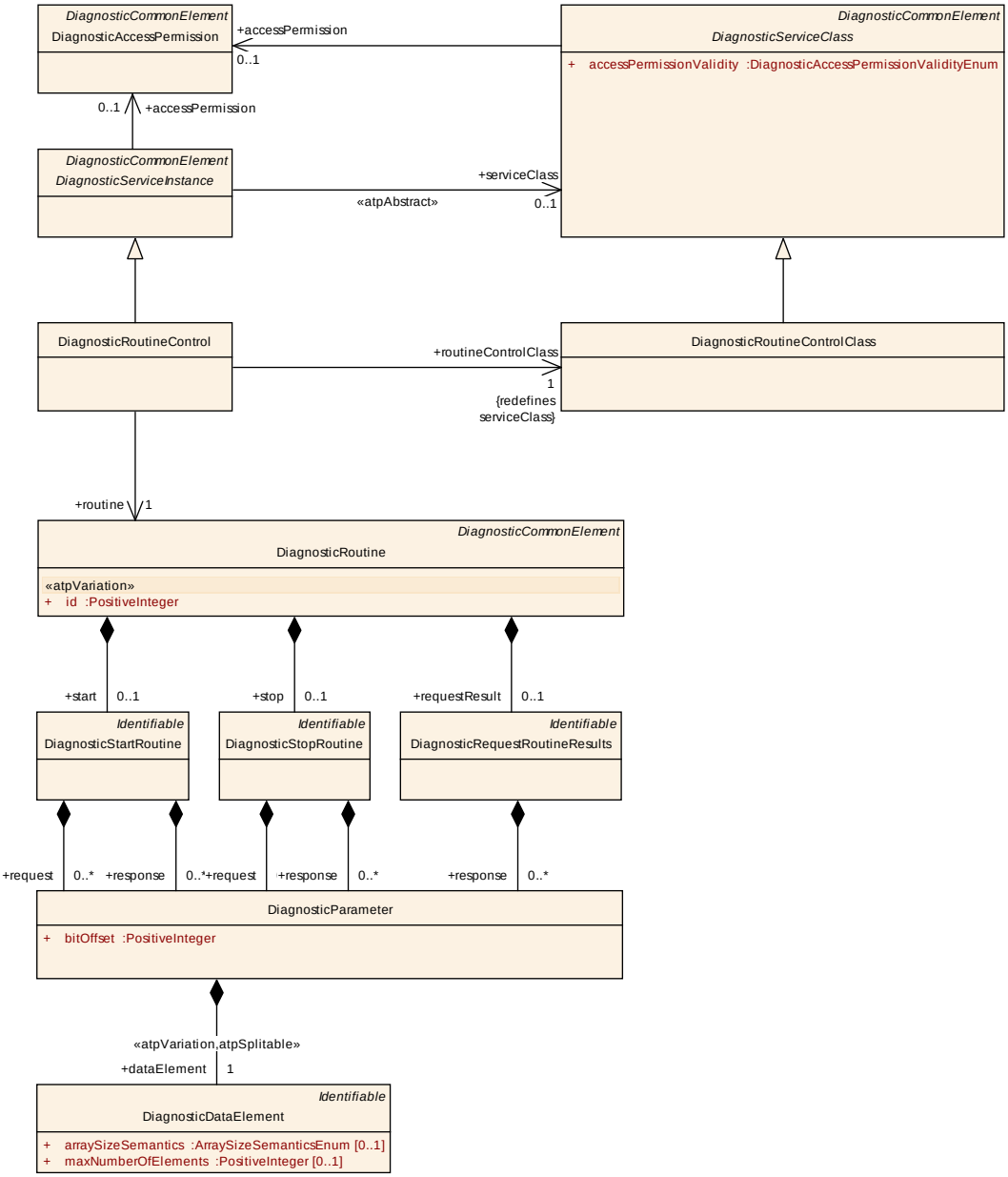

Figure 5.20: Modeling of diagnostic service `RoutineControl` (0x31)

[TPS_DEXT_01079] Modeling of the arguments to a DiagnosticRoutine ⌈ The arguments to a DiagnosticRoutine are modeled by means of DiagnosticParameter that is aggregated in the following roles:

- DiagnosticStartRoutine.request
- DiagnosticStartRoutine.response
- DiagnosticStopRoutine.request
- DiagnosticStopRoutine.response
- DiagnosticRequestRoutineResults.response

⌋*(RS_DEXT_00015)*

A DiagnosticParameter, in turn, aggregates a DiagnosticDataElement (see section 4.2) in the role dataElement.

[TPS_DEXT_01080] Diagnostic Routine needs to be started ⌈ ISO 14229-1 [15] does not foresee the existence of a Diagnostic Routine that is already executing at boot time. Therefore, a Diagnostic Routine needs to be started at some point in order to make sense of it. ⌋*(RS_DEXT_00015)*

[constr_1339] Existence of DiagnosticRoutine.start ⌈ In a complete DiagnosticExtract, the attribute DiagnosticRoutine.start shall always exist for any given DiagnosticRoutine. ⌋*()*

[TPS_DEXT_01035] Existence of DiagnosticRoutine.stop and DiagnosticRoutine.requestResult ⌈ In contrast to DiagnosticRoutine.start (as clarified by [constr_1339]), the existence of DiagnosticRoutine.stop and DiagnosticRoutine.requestResult is truly optional. ⌋*(RS_DEXT_00015)*

[constr_1340] Consistency of DiagnosticServiceSwMapping with respect to synchronously called DiagnosticRoutines ⌈ Each DiagnosticServiceSwMapping that references a DiagnosticRoutineControl that only aggregates a DiagnosticStartRoutine in the role start shall only reference a SwcServiceDependency or BswServiceDependency that in turn aggregates a DiagnosticRoutineNeeds with attribute diagRoutineType set to DiagnosticRoutineTypeEnum.synchronous. ⌋*()*

[constr_1341] Consistency of DiagnosticServiceSwMapping with respect to asynchronously called DiagnosticRoutines ⌈ Each DiagnosticServiceSwMapping that references a DiagnosticRoutineControl that aggregates a DiagnosticStopRoutine and/or DiagnosticRequestRoutineResults in the role stop resp. requestResults shall only reference a SwcServiceDependency or BswServiceDependency that in turn aggregates a DiagnosticRoutineNeeds with attribute diagRoutineType set to DiagnosticRoutineTypeEnum.asynchronous. ⌋*()*

[TPS_DEXT_01049] Consistency of DiagnosticServiceSwMapping with respect to routine IDs ⌈ For each DiagnosticServiceSwMapping that references

a DiagnosticRoutineNeeds and a DiagnosticRoutineControl, the value of DiagnosticRoutineNeeds.ridNumber shall be ignored and the value of DiagnosticRoutineControl.routine.id shall be taken instead. ⌋(*RS_DEXT_00015*, *RS_DEXT_00052*)

Class	**DiagnosticRoutine**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::CommonDiagnostics			
Note	This meta-class represents the ability to define a diagnostic routine.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
id	PositiveInteger	1	attr	This is the numerical identifier used to identify the DiagnosticRoutine in the scope of diagnostic workflow **Stereotypes:** atpVariation **Tags:** vh.latestBindingTime=preCompileTime
requestResult	DiagnosticRequestRoutineResults	0..1	aggr	This represents the ability to request the result of a running routine.
start	DiagnosticStartRoutine	0..1	aggr	This represents the ability to start a routine
stop	DiagnosticStopRoutine	0..1	aggr	This represents the ability to stop a running routine.

Table 5.56: DiagnosticRoutine

Class	**DiagnosticStartRoutine**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::CommonDiagnostics			
Note	This represents the ability to start a diagnostic routine.			
Base	ARObject,Identifiable,MultilanguageReferrable,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
request	DiagnosticParameter	*	aggr	This represents the request parameters.
response	DiagnosticParameter	*	aggr	This represents the response parameters.

Table 5.57: DiagnosticStartRoutine

Class	**DiagnosticStopRoutine**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::CommonDiagnostics			
Note	This represents the ability to stop a diagnostic routine.			
Base	ARObject,Identifiable,MultilanguageReferrable,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
request	DiagnosticParameter	*	aggr	This represents the request parameters.
response	DiagnosticParameter	*	aggr	This represents the respone parameters.

Attribute	Datatype	Mul.	Kind	Note

Table 5.58: DiagnosticStopRoutine

Class	DiagnosticRequestRoutineResults			
Package	M2::AUTOSARTemplates::DiagnosticExtract::CommonDiagnostics			
Note	This meta-class represents the ability to define the result of a diagnostic routine execution.			
Base	ARObject,Identifiable,MultilanguageReferrable,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
response	DiagnosticParameter	*	aggr	This represents the response parameters.

Table 5.59: DiagnosticRequestRoutineResults

Class	DiagnosticRoutineControl			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::RoutineControl			
Note	This represents an instance of the "Routine Control" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticServiceInstance,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
routine	DiagnosticRoutine	1	ref	This refers to the applicable DiagnosticRoutine.
routineControlClass	DiagnosticRoutineControlClass	1	ref	This reference substantiates that abstract reference in the role serviceClass for this specific concrete class. Thereby, the reference represents the ability to access shared attributes among all DiagnosticRoutineControl in the given context.

Table 5.60: DiagnosticRoutineControl

5.4.13 SecurityAccess

This chapter describes the modeling of diagnostic services `SecurityAccess` (0x27).

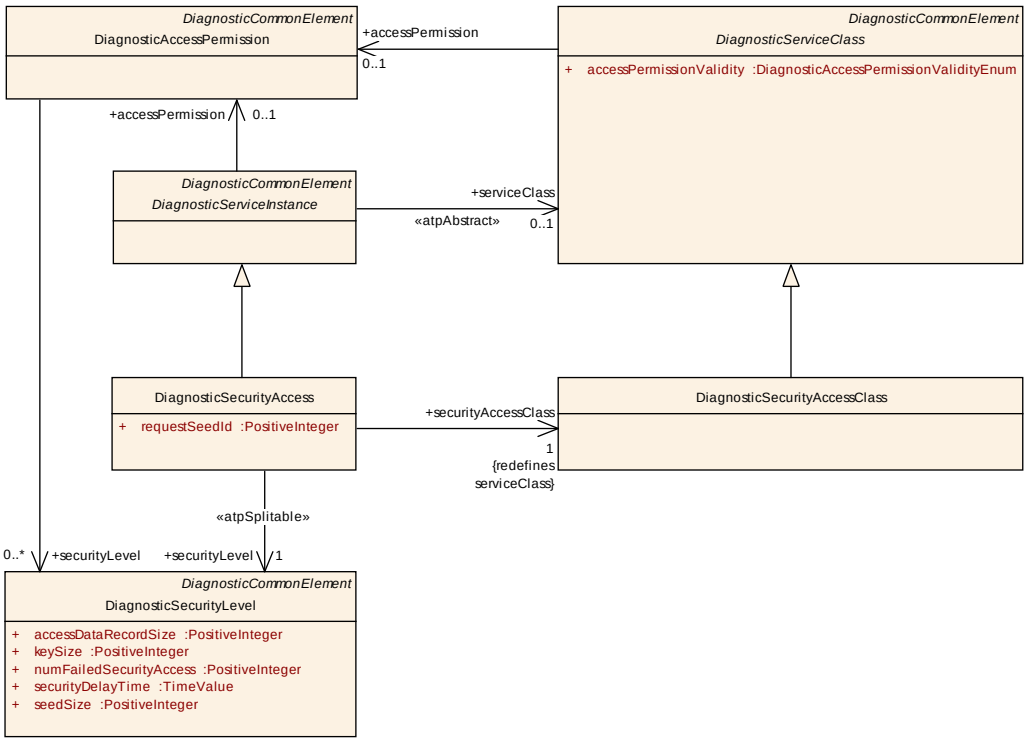

Figure 5.21: Modeling of diagnostic service `SecurityAccess` (0x27)

[TPS_DEXT_01053] Existence of `DiagnosticSecurityAccess.securityLevel`
⌈ The configuration of a given `DiagnosticSecurityAccess` is considered incomplete until the reference in the role `DiagnosticSecurityAccess.securityLevel` exists. ⌋*(RS_DEXT_00009, RS_DEXT_00042)*

The meaning of [TPS_DEXT_01053] is that the reference may be missing in intermediate steps of the configuration work flow. But at the point in time where ECU configuration is generated from the `DiagnosticExtract` the reference is needed to able to make sense of the model for the given `DiagnosticSecurityAccess`.

Please note that (as already explained in section 5.4) the sub-functions of this service are modeled by means of the `category` attribute.

In response to conceptual differences between many other diagnostic services and `SecurityAccess`, the applicable sub-functions for the diagnostic service `SecurityAccess` are **not** defined by means of the attribute `DiagnosticSecurityAccess.category`.

[TPS_DEXT_01036] Work-flow within the execution of the diagnostic service `SecurityAccess` ⌈ The work-flow within the execution of the diagnostic service `SecurityAccess` basically boils down to the tester sending the request to obtain a *seed* from the diagnostic stack and then sending back a *key* to the stack.

Thus, the sub-functions could be generically be described as *requestSeed* and *sendKey*, which is precisely what the ISO 14229-1 [15] does.

According to this logic, the *requestSeed* could get a specific number assigned to identify the sub-function and then the *sendKey* sub-function would get assigned the number of the *requestSeed* sub-function + 1. Again, this is fully in line with the ISO 14229-1 [15]. ⌋*(RS_DEXT_00009)*

However, there is further dimension to take into account, namely the DiagnosticSecurityLevel. According to ISO 14229-1 [15], different security levels make different numbers for the sub-function identifier.

[TPS_DEXT_01037] Semantics of DiagnosticSecurityAccess.requestSeedId ⌈ The attribute DiagnosticSecurityAccess.requestSeedId shall be used to define the number of the sub-function of the diagnostic service SecurityAccess according to the intended security level. ⌋*(RS_DEXT_00009)*

[constr_1342] Possible values for DiagnosticSecurityAccess.requestSeedId ⌈ The value of the attribute DiagnosticSecurityAccess.requestSeedId **shall only be set to an odd number**[1].

The supported value range consists of the following list:

- all odd numbers in the closed interval **0x01** .. **0x41**
- **0x5F** (this corresponds to the case of *end-of-life activation of on-board pyrotechnic devices according to ISO 26021-2 [16]*)
- all odd numbers in the closed interval **0x61** .. **0x7E**

⌋*()*

In contrast to a similar situation in the case of the diagnostic service SessionControl (see section 5.4.14), there is no real evidence that a DiagnosticSecurityLevel always exists before the referencing DiagnosticSecurityAccess is created in order to properly establish the reference in the role DiagnosticSecurityAccess.securityLevel.

[TPS_DEXT_01038] Motivation for making the reference DiagnosticSecurityAccess.securityLevel ≪atpSplitable≫ ⌈ The reference DiagnosticSecurityAccess.securityLevel needs to be decorated with the stereotype ≪atpSplitable≫ in order to advertise the idea that the reference to a corresponding DiagnosticSecurityLevel is created (potentially in a different artifact) some time after the actual creation of the given DiagnosticSecurityAccess. ⌋*(RS_DEXT_00002, RS_DEXT_00009, RS_DEXT_00042)*

Of course, if the DiagnosticSecurityLevel factually exists before the definition of DiagnosticSecurityAccess the reference can directly be inserted into the model.

[1] The even numbers are reserved for the identification of the corresponding *sendKey* sub-function, as explained by [TPS_DEXT_01036].

Class	**DiagnosticSecurityAccess**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::Security Access			
Note	This represents an instance of the "Security Access" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Diagnostic ServiceInstance,Identifiable,MultilanguageReferrable,Packageable Element,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
requestSeedId	PositiveInteger	1	attr	This would be 0x01, 0x03, 0x05, ... The sendKey id can be computed by adding 1 to the requestSeedId
securityAccessClass	DiagnosticSecurityAccessClass	1	ref	This reference substantiates that abstract reference in the role serviceClass for this specific concrete class. Thereby, the reference represents the ability to access shared attributes among all DiagnosticSecurityAccess in the given context.
securityLevel	DiagnosticSecurityLevel	1	ref	**Stereotypes:** atpSplitable**Tags:** atp.Splitkey=securityLevel

Table 5.61: DiagnosticSecurityAccess

5.4.14 SessionControl

This chapter describes the modeling of diagnostic services `SessionControl` (0x10). The obvious goal of the service is to support the switching from one diagnostic session to another.

[TPS_DEXT_01081] Modeling of `DiagnosticSessionControl` ⌈ For the purpose of providing a means to specify the switching from one diagnostic session to another diagnostic session, `DiagnosticSessionControl` refers to a `DiagnosticSession` in the role `diagnosticSession`. ⌋*(RS_DEXT_00003, RS_DEXT_00040)*

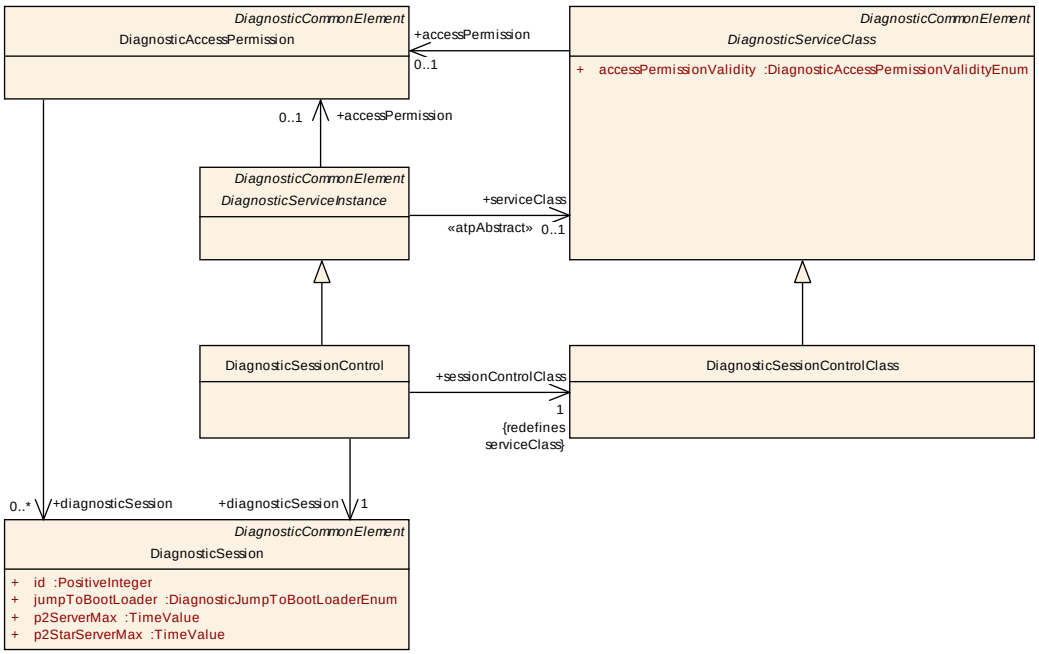

Figure 5.22: Modeling of diagnostic service `SessionControl` (0x10)

According to ISO 14229-1 [15], the diagnostic service `SessionControl` defines sub-functions.

[TPS_DEXT_01039] Identification of the sub-function of `DiagnosticSessionControl` ⌈ In the case of `DiagnosticSessionControl` it would not be a good idea to encode the applicable sub-function by means of the attribute `DiagnosticSessionControl.category`.

Actually, the possible sub-functions are strongly related to the concept of the diagnostic session, represented by the meta-class `DiagnosticSession`.

The latter, in turn, has an attribute `id` that directly corresponds to the number of the applicable sub-function for `DiagnosticSessionControl`.

In other words, the sub-function of `DiagnosticSessionControl` is identified by means of the reference `DiagnosticSessionControl.diagnosticSession`. ⌋*(RS_DEXT_00003, RS_DEXT_00051)*

[TPS_DEXT_01082] Existence of `DiagnosticSessionControl.diagnosticSession` ⌈ The idea of modeling the sub-function of `DiagnosticSessionControl` by means of the reference `DiagnosticSessionControl.diagnosticSession` implies that the applicable `DiagnosticSession` already exists at the time when the given `DiagnosticSessionControl` is created.

It is assumed that this will always be the case because the definition of `DiagnosticSessions` is part of laying the groundwork[2] for diagnostic communication. ⌋*(RS_DEXT_00003, RS_DEXT_00040)*

[2]This is similar to the definition of commonly used data types in a software development project

It is hard to foresee a scenario where the DiagnosticSessions are defined near the very end of the work-flow that leads to a complete DiagnosticExtract.

Class	**DiagnosticSessionControl**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::SessionControl			
Note	This represents an instance of the "Session Control" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticServiceInstance,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
diagnosticSession	DiagnosticSession	1	ref	This represents the applicable DiagnosticSessions
sessionControlClass	DiagnosticSessionControlClass	1	ref	This reference substantiates that abstract reference in the role serviceClass for this specific concrete class. Thereby, the reference represents the ability to access shared attributes among all DiagnosticSessionControl in the given context.

Table 5.62: DiagnosticSessionControl

5.4.15 RequestFileTransfer

This chapter describes the modeling of diagnostic services RequestFileTransfer (0x38). The purpose of the service is the triggering of the transfer of a *file* from or to the AUTOSAR diagnostic stack.

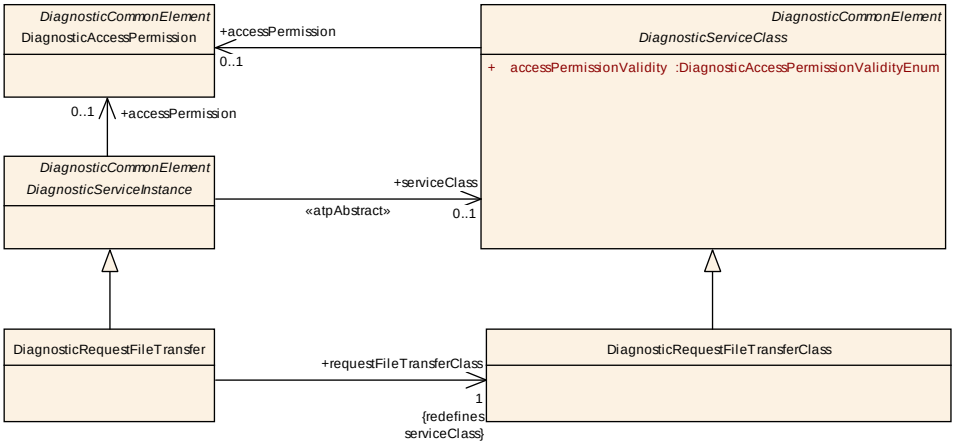

Figure 5.23: Modeling of diagnostic service RequestFileTransfer (0x38)

Please note that there is nothing to configure for DiagnosticRequestFileTransfer beyond its mere existence.

[TPS_DEXT_01090] Diagnostic service RequestFileTransfer does not define any sub-functions ⌈ The diagnostic service RequestFileTransfer does not define

any sub-functions. therefore, the usage of the attribute `category` is not constrained for meta-class `DiagnosticRequestFileTransfer`. ⌋(*RS_DEXT_00057*)

Class	**DiagnosticRequestFileTransfer**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::RequestFile Transfer			
Note	This diagnostic service instance implements the UDS service 0x38.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Diagnostic ServiceInstance,Identifiable,MultilanguageReferrable,Packageable Element,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
requestFile TransferCl ass	DiagnosticRequ estFileTransfer Class	1	ref	This reference substantiates that abstract reference in the role serviceClass for this specific concrete class. Thereby, the reference represents the ability to access shared attributes among all DiagnosticRequestFileTransfer in the given context.

Table 5.63: DiagnosticRequestFileTransfer

Class	**DiagnosticRequestFileTransferClass**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm::DiagnosticService::RequestFile Transfer			
Note	This meta-class contains attributes shared by all instances of the "Request File transfer" diagnostic service.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Diagnostic ServiceClass,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
–	–	–	–	–

Table 5.64: DiagnosticRequestFileTransferClass

5.5 Diagnostic Service Mapping

Automotive diagnostics interacts with both application software and basic software in various ways that can be formalized using the AUTOSAR meta-model. This chapter contains a description of the formalization of this interaction along with the applicable constraints.

[TPS_DEXT_01040] Use case where the `DiagnosticExtract` refers to software-components ⌈ This is a list of the potential use case where the `DiagnosticExtract` refers to software-components in general and `PortPrototype`s in the context of either `CompositionSwComponentType`s or `AtomicSwComponentType`s:

- DiagnosticExtract refers to PortPrototype (for the access to dataElement) or SwcServiceDependency in the context of a AtomicSwComponentType embedded in the hierarchy created by the rootSoftwareComposition.
- DiagnosticExtract refers to a PortPrototype(for the access to dataElement) or SwcServiceDependency in the context of a AtomicSwComponentType embedded in the hierarchy created by a CompositionSwComponentType that is nowhere aggregated (for the time being).
- DiagnosticExtract refers to a PortPrototype (for the access to dataElement) or SwcServiceDependencyin the context of an AtomicSwComponentType.
- DiagnosticExtract refers to a BswServiceDependency.

⌋(*RS_DEXT_00052*)

5.5.1 Diagnostic Service Data Mapping

Please note that the Dcm is in general entitled to both read and write a dataElement. This applies even if the corresponding PortPrototype is a PPortPrototype. This means that the diagnostic service data mapping is limited to SenderReceiverInterface.

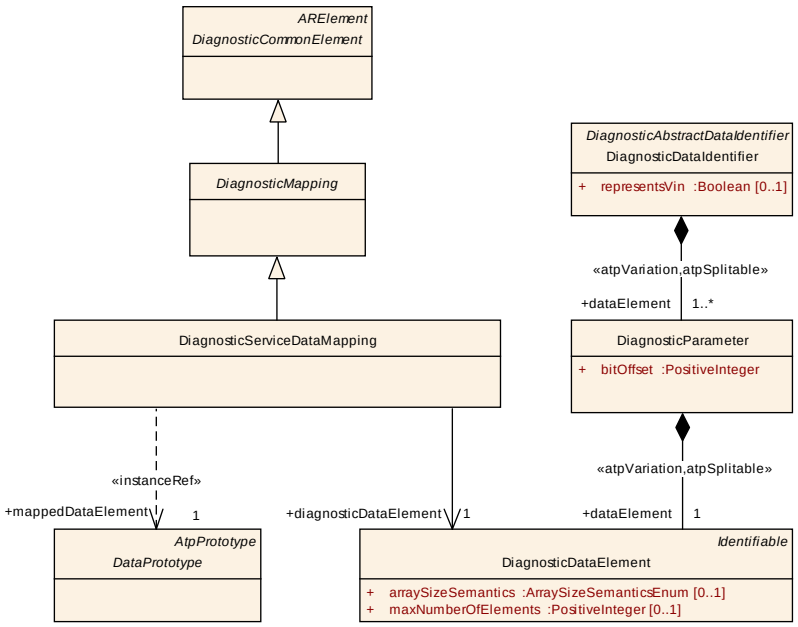

Figure 5.24: Dcm service data mapping

[TPS_DEXT_01041] Semantics of attribute DiagnosticServiceDataMapping.diagnosticDataElement ⌈ By means of the attribute DiagnosticServiceDataMapping.diagnosticDataElement it is possible to specify that the Dcm has access to a dataElement in a PortPrototype typed by a SenderReceiverInterface.

This type of data access is suitable for the diagnostic services ReadDataByIdentifier (0x22) and WriteDataByIdentifier (0x2E). ⌋(*RS_DEXT_00052*)

[constr_1343] Simultaneous existence of the attributes DiagnosticServiceDataMapping.diagnosticDataElement and DiagnosticDataByIdentifier.dataIdentifier ⌈ A DiagnosticServiceDataMapping.diagnosticDataElement shall also be aggregated by a DiagnosticDataByIdentifier in the role dataIdentifier.dataElement.dataElement. ⌋()

Please note that [constr_1343] shall only apply for the step in the methodology where the DiagnosticExtract is considered complete to the point that the configuration of the Dcm and Dem can be derived. Any intermediate step, e.g. hand-over from OEM to tier-1 supplier does not actually enforce [constr_1343].

In other words, [constr_1343] makes sure that there is a connection between the DiagnosticServiceDataMapping and the corresponding DiagnosticReadDataByIdentifier or DiagnosticWriteDataByIdentifier.

Only by this means the diagnostic service becomes fully usable.

[constr_1344] Condition for the identification of data types of attributes DiagnosticServiceDataMapping.mappedDataElement and DiagnosticServiceDataMapping.diagnosticDataElement ⌈ Both DiagnosticServiceDataMapping.mappedDataElement and DiagnosticServiceDataMapping.diagnosticDataElement shall be typed by either of the following options:

- ApplicationPrimitiveDataType where the value of attribute category is set to VALUE.
- ImplementationDataType where the value of attribute category is set to VALUE or to TYPE_REFERENCE that eventually resolves to an ImplementationDataType where attribute category is set to VALUE.

⌋()

In other words, [constr_1344] requires that both DiagnosticServiceDataMapping.mappedDataElement[3] and DiagnosticServiceDataMapping.diagnosticDataElement shall be typed by a primitive data type.

Please refer to [9] for a detailed explanation of the meaning of the value of a data type category.

[3] DiagnosticServiceDataMapping.mappedDataElement can be an element of a (potentially large) composite data type. The utility of this is that this way the footprint of the data access to the payload of request and response messages can be kept as low as possible.

Class	**DiagnosticServiceDataMapping**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::ServiceMapping			
Note	This represents the ability to define a mapping of a diagnostic service to a software-component. This kind of service mapping is applicable for the usage of SenderReceiverInterfaces.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticMapping,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
diagnosticDataElement	DiagnosticDataElement	1	ref	This represents the applicable payload that corresponds to the referenced DataPrototype in the role mappedDataElement.
mappedDataElement	DataPrototype	1	iref	This represents the dataElement in the application software that is accessed for diagnostic purpose.

Table 5.65: DiagnosticServiceDataMapping

[constr_1345] `DiagnosticDataElement` shall not (finally) be aggregated by a `DiagnosticRoutine` ⌈ A `DiagnosticDataElement` that is referenced by a `DiagnosticServiceDataMapping` shall not (finally) be aggregated by a `DiagnosticRoutine`. ⌋*()*

[TPS_DEXT_01042] Dem uses `DiagnosticServiceDataMapping` ⌈ There is a use case for the Dem to utilize a `DiagnosticServiceDataMapping` such that elements of a `DiagnosticExtendedDataRecord` are fetched from `dataElement`s in an `ApplicationSwComponentType`.

Therefore, [constr_1345] does intentionally not exclude the aggregation of `DiagnosticDataElement` by `DiagnosticExtendedDataRecord` in the context of `DiagnosticServiceDataMapping`. ⌋*(RS_DEXT_00052)*

5.5.2 Diagnostic Service Software Mapping

The diagnostic service software mapping is limited to `ClientServerInterface` or a direct function call (in the case of basic software or complex driver).

[TPS_DEXT_01043] Purpose of `DiagnosticServiceSwMapping` ⌈ The meta-class `DiagnosticServiceSwMapping` has been introduced to support the creation of a relationship between the definition of a given diagnostic service to the `SwcServiceDependency` (if the service applies to the application software) or `BswServiceDependency` (if the service applies to the basic software). ⌋*(RS_DEXT_00052)*

It is required to use the applicable form of reference to the target `SwcServiceDependency` depending on the context of the enclosing `AtomicSwComponentType`.

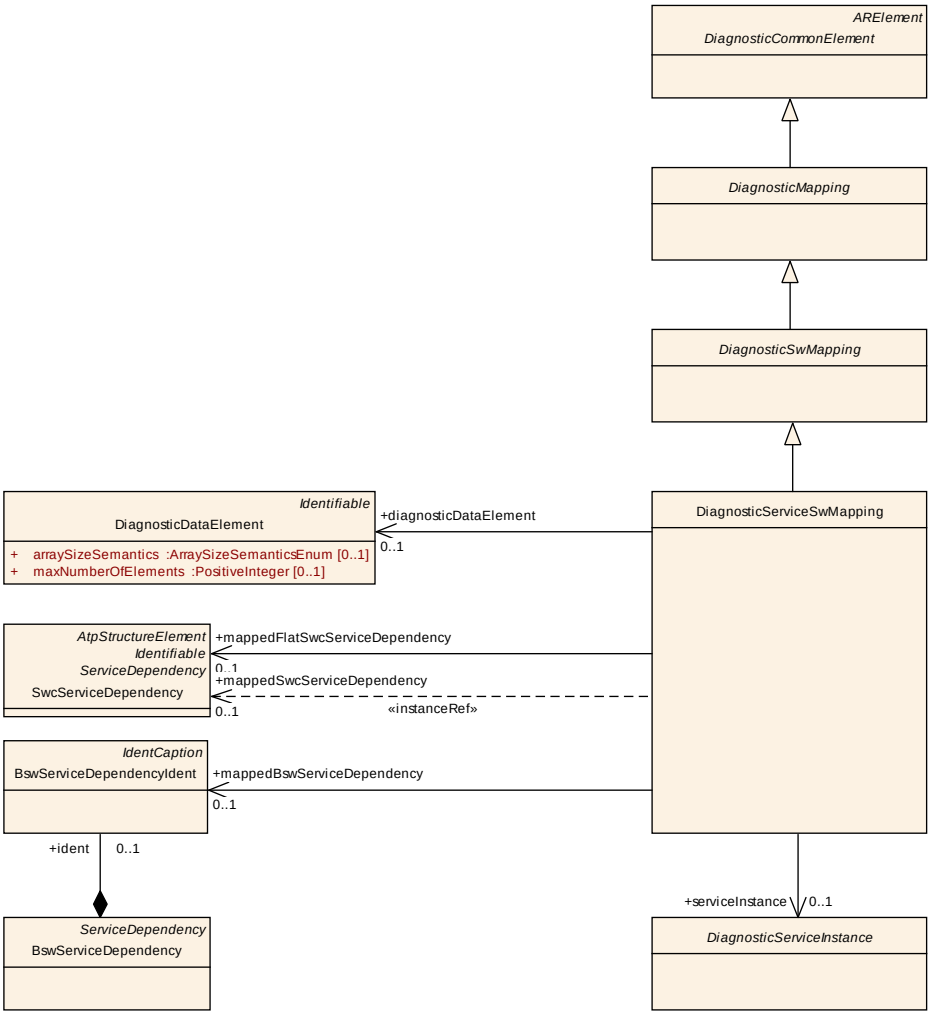

Figure 5.25: Dcm service software mapping

Class	**DiagnosticSwMapping (abstract)**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::ServiceMapping			
Note	This represents the ability to define a mapping between a diagnostic information (at this point there is no way to become more specific about the semantics) to a software-component.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticMapping,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
–	–	–	–	–

Table 5.66: DiagnosticSwMapping

Class	**DiagnosticServiceSwMapping**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::ServiceMapping			
Note	This represents the ability to define a mapping of a diagnostic service to a software-component. This kind of service mapping is applicable for the usage of ClientServerInterfaces.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticMapping,DiagnosticSwMapping,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
diagnosticDataElement	DiagnosticDataElement	0..1	ref	This represents a DiagnosticDataElement required to execute the respective diagnostic service in the context of the diagnostic service mapping,
mappedBswServiceDependency	BswServiceDependencyIdent	0..1	ref	This is supposed to represent a reference to a BswServiceDependency. the latter is not derived from Referrable and therefore this detour needs to be implemented to still let BswServiceDependency become the target of a reference.
mappedFlatSwcServiceDependency	SwcServiceDependency	0..1	ref	This represents the ability to refer to an AtomicSwComponentType that is available without the definition of how it will be emebdded into the component hiearchy.
mappedSwcServiceDependency	SwcServiceDependency	0..1	iref	This represents the ability to point into the component hiearchy (under possible consideration of the rootSoftwareComposition)
serviceInstance	DiagnosticServiceInstance	0..1	ref	This represents the service instance that needs to be considered in this diagnostics service mapping,

Table 5.67: DiagnosticServiceSwMapping

[TPS_DEXT_01044] `BswServiceDependency` needs to act as the target of a reference ⌈ As indicated by Figure 5.25, the intention of `DiagnosticServiceSwMapping.mappedBswServiceDependency` is to refer to a `BswServiceDependency` in the same way as e.g. `DiagnosticServiceSwMapping.mappedFlatSwcServiceDependency` does.

However, `BswServiceDependency` is not derived from meta-class `Referrable` and can therefore never become the target of a reference like `DiagnosticServiceSwMapping.mappedBswServiceDependency`.

The remedy for this issue is to define meta-class `BswServiceDependencyIdent` that inherits from `IdentCaption` that in turn inherits from `Referrable`.

Then, by aggregating `BswServiceDependencyIdent` at `BswServiceDependency` in the role `ident BswServiceDependency` can **factually** become the target of the reference and thus the original idea of `DiagnosticServiceSwMapping.mappedBswServiceDependency` becomes feasible. ⌋*(RS_DEXT_00052)*

Please note that the introduction [TPS_DEXT_01044], although being dangerously close to a hack, is necessary to keep the AUTOSAR XML Schema fully backwards-compatible.

In other words, if BswServiceDependency were updated to inherit from Referrable the consequence would be that all existing AUTOSAR models that contain instances BswServiceDependency would suddenly become invalid because Referrable.shortName is a mandatory attribute in the AUTOSAR XML Schema.

Class	**BswServiceDependencyIdent**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::ServiceMapping			
Note	This meta-class is created to add the ability to become the target of a reference to the non-Referrable BswServiceDependency.			
Base	ARObject,AtpClassifier,AtpFeature,AtpStructureElement,Ident Caption,Identifiable,MultilanguageReferrable,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
–	–	–	–	–

Table 5.68: BswServiceDependencyIdent

[constr_1346] Allowed values of DiagnosticServiceSwMapping.serviceInstance ⌈ The applicability of the DiagnosticServiceSwMapping is limited to predefined set of diagnostic services.

By regulation of the AUTOSAR standard, DiagnosticServiceSwMapping.serviceInstance shall only point to the following sub-classes of DiagnosticServiceInstance:

- DiagnosticRoutine
- DiagnosticSecurityAccess
- DiagnosticReadDataByIdentifier
- DiagnosticWriteDataByIdentifier
- DiagnosticIOControl

⌋()

[constr_1347] Existence of attributes of DiagnosticServiceSwMapping ⌈ For any given DiagnosticServiceSwMapping, **one and only one of** the following references shall exist:

- DiagnosticServiceSwMapping.mappedFlatSwcServiceDependency
- DiagnosticServiceSwMapping.mappedSwcServiceDependency
- DiagnosticServiceSwMapping.mappedBswServiceDependency

⌋()

[constr_1347], among further clarifications, reflects the fact that at most a single SwcServiceDependency can be referenced by a DiagnosticServiceSwMapping and this SwcServiceDependency cannot be identified by both mappedSwcServiceDependency and mappedFlatSwcServiceDependency.

6 Diagnostic Event Handling

6.1 Introduction

This subchapter describes the meta-model elements that define the handling of and the functionality around diagnostic events.

In a standard AUTOSAR Basic Software architecture, the definitions based on the model elements described in this subchapter are realized by the Diagnostic Event Manager (Dem) module.

The following figure gives an overview on the model elements related to the diagnostic event functionality.

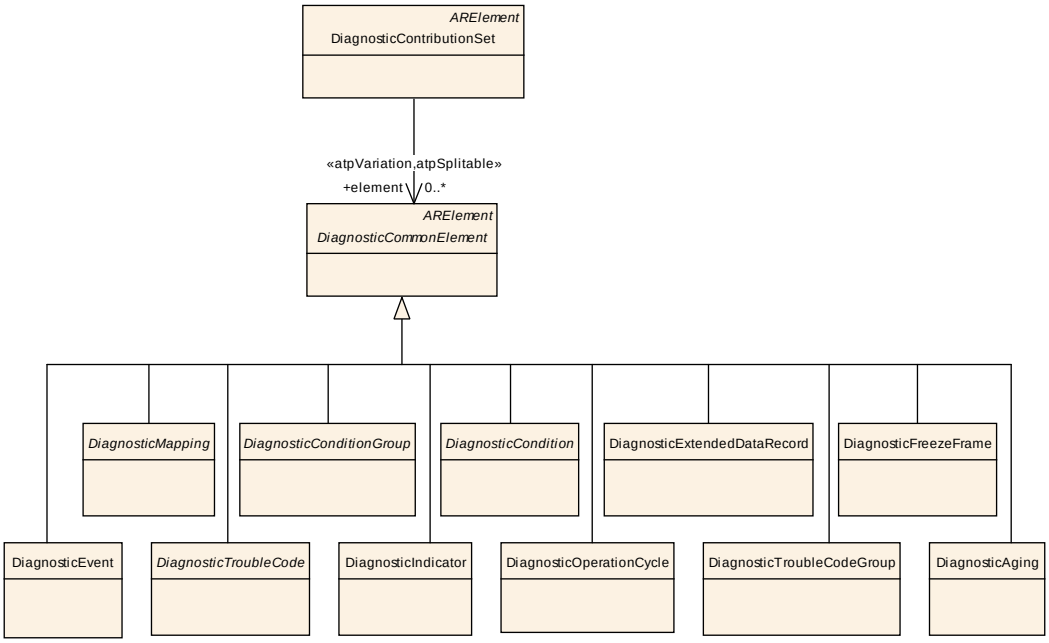

Figure 6.1: Common model elements relevant for the Dem

For the definition of the diagnostic event functionality, a number of model elements are derived from DiagnosticCommonElement. These elements are described in the following sub-chapters.

6.2 DiagnosticEvent

[TPS_DEXT_01083] Semantics of a DiagnosticEvent ⌈ A DiagnosticEvent - the atomic unit handled by the Dem module - has to be defined together with its properties which affect the event handling behavior and possible interfaces to software-components. ⌋*(RS_DEXT_00023)*

Figure 6.2 depicts the definition of DiagnosticEvent together with its properties.

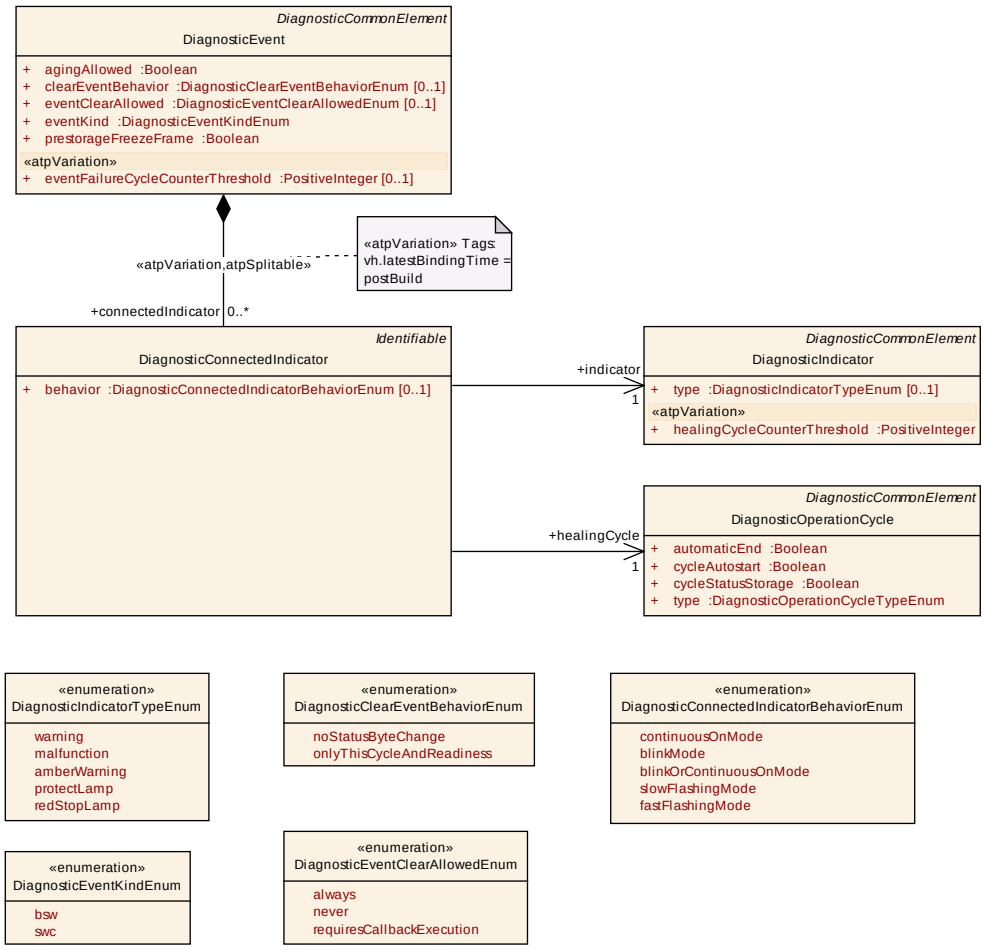

Figure 6.2: Modelling of DiagnosticEvent

The DiagnosticExtract allows the definition of an arbitrary number of DiagnosticEvents.

Although the exchange of a DiagnosticExtract between companies usually involves DiagnosticEvents related to SWC functionality, the event kind BSW is also supported in order to enable definition of handling of BSW events (e.g. definition of associated DiagnosticTroubleCode).

[TPS_DEXT_03011] Clearing request for a DiagnosticEvent ⌈ Furthermore, a clearing request for a DiagnosticEvent might require invocation of a callback to a SWC in order to allow or to prohibit the clearing operation.

The expectation on this callback interface can be expressed using the attribute eventClearAllowed:

- always indicates that a clearing request for the DiagnosticEvent **shall unconditionally be executed**.
- never denotes that a clearing for the DiagnosticEvent is **intentionally not possible**.

- In case of `requiresCallbackExecution`, the **execution of a callback shall decide whether the clearing is permitted**.

  In other words, the implementation of this decision is up to the developer of the corresponding `AtomicSwComponentType`.

  The latter shall define a `SwcServiceDependency` with appropriate `DiagnosticEventNeeds` and a `RoleBasedPortAssignment` where the value of the attribute `role` is set to `CallbackClearEventAllowed`.

⌋(*RS_DEXT_00023*)

Class	**DiagnosticEvent**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticEvent			
Note	This element is used to configure DiagnosticEvents.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
agingAllowed	Boolean	1	attr	This represents the decision whether aging is allowed for this DiagnosticEvent.
clearEventBehavior	DiagnosticClearEventBehaviorEnum	0..1	attr	This attribute defines the resulting UDS status byte for the related event, which shall not be cleared according to the ClearEventAllowed callback.
connectedIndicator	DiagnosticConnectedIndicator	*	aggr	Event specific description of Indicators. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=shortName, variationPoint.shortLabel vh.latestBindingTime=postBuild
eventClearAllowed	DiagnosticEventClearAllowedEnum	0..1	attr	This attribute defines whether the Dem has access to a "ClearEventAllowed" callback.
eventFailureCycleCounterThreshold	PositiveInteger	0..1	attr	This attribute defines the number of failure cycles for the event based fault confirmation. **Stereotypes:** atpVariation **Tags:** vh.latestBindingTime=postBuild
eventKind	DiagnosticEventKindEnum	1	attr	This attribute is used to distinguish between SWC and BSW events.
prestorageFreezeFrame	Boolean	1	attr	This attribute describes whether the Prestorage of FreezeFrames is supported by the assigned event or not. True: Prestorage of FreezeFrames is supported False: Prestorage of FreezeFrames is not supported

Table 6.1: DiagnosticEvent

[TPS_DEXT_01085] `DiagnosticEvent` can be connected to one or multiple indicators ⌈ A `DiagnosticEvent` can be connected to one or multiple indicators (mod-

eled by means of aggregating DiagnosticIndicator in the role connectedIndicator) of a certain type and with certain behavior. ⌋(*RS_DEXT_00023*)

[TPS_DEXT_01067] Textually formulated content attached to DiagnosticEvent ⌈ The definition of a DiagnosticEvent also consists of textually formulated content that is formalized in structure but cannot be formalized in content.

The purpose of this content is to define e.g. a mature condition that relates to the specific DiagnosticEvent. ⌋(*RS_DEXT_00023, RS_DEXT_00045*)

[TPS_DEXT_01068] Textual description with respect to the DiagnosticEvent ⌈ Textual description that has the character of requirements with respect to the DiagnosticEvent shall be provided by means of the meta-class StructuredReq, i.e. by means of introduction.structuredReq. ⌋(*RS_DEXT_00023, RS_DEXT_00045*)

For more details regarding the modeling of the semi-formal text please refer to Figure 4.3.

[TPS_DEXT_01069] Standardized values of DiagnosticEvent.introduction.structuredReq ⌈ The following possible values of DiagnosticEvent.introduction.struturedReq are standardized by AUTOSAR:

- **DIAG_EVENT_MON_COND**: this value describes the monitoring condition of the corresponding DiagnosticEvent.
- **DIAG_EVENT_MON_TYPE**: this value describes the monitoring type of the corresponding DiagnosticEvent.
- **DIAG_EVENT_MON_RATE**: this value describes the monitoring rate for the corresponding DiagnosticEvent.
- **DIAG_EVENT_MAT_COND**: this value describes a mature condition of the DiagnosticEvent.
- **DIAG_EVENT_DEMAT_COND**: this value describes a de-mature condition of the DiagnosticEvent.
- **DIAG_EVENT_AGING**: this value describes the behavior of the DiagnosticEvent regarding aging.
- **DIAG_EVENT_LIMP_IN_ACT**: this value describes the associated limp-in action for the DiagnosticEvent.
- **DIAG_EVENT_MAT_TIME**: this value describes the mature time for the corresponding DiagnosticEvent, i.e. how long or how often the fault must exist.
- **DIAG_EVENT_DEMAT_TIME**: this value describes the de-mature time for the corresponding DiagnosticEvent, i.e. how long or how often must the OK conditions be fulfilled.

⌋(*RS_DEXT_00001, RS_DEXT_00023, RS_DEXT_00045*)

The following ARXML fragment exemplifies the usage of StructuredReq along with the standardized values of the attribute category to attach semi-formal textual descriptions to a DiagnosticEvent.

Listing 6.1: Example for the definition of a semi-formal textual elements in the context of a DiagnosticEvent

```
<DIAGNOSTIC-EVENT>
    <SHORT-NAME>ExampleEvent_0001</SHORT-NAME>
        <INTRODUCTION>
            <STRUCTURED-REQ>
                <SHORT-NAME>MatureCondition</SHORT-NAME>
                <CATEGORY>DIAG_EVENT_MAT_COND</CATEGORY>
                <DESCRIPTION>
                    <P>
                        <L-1 L="EN">This DTC is set if System Voltage is
                            below 9 Volts</L-1>
                    </P>
                </DESCRIPTION>
            </STRUCTURED-REQ>
            <STRUCTURED-REQ>
                <SHORT-NAME>DematureCondition</SHORT-NAME>
                <CATEGORY>DEMATURE_COND</CATEGORY>
                <DESCRIPTION>
                    <P>
                        <L-1 L="EN">This DTC is set if System Voltage is
                            above 10 Volts<XFILE><SHORT-NAME>
                            Requirement_Specification</SHORT-NAME><URL>http:
                            //autosar.org</URL></XFILE>
                        </L-1>
                    </P>
                </DESCRIPTION>
            </STRUCTURED-REQ>
        </INTRODUCTION>
    <CLEAR-EVENT-BEHAVIOR>ONLY-THIS-CYCLE-AND-READINESS</CLEAR-EVENT-
        BEHAVIOR>
    <EVENT-CLEAR-ALLOWED>ALWAYS</EVENT-CLEAR-ALLOWED>
    <EVENT-FAILURE-CYCLE-COUNTER-THRESHOLD>100</EVENT-FAILURE-CYCLE-COUNTER
        -THRESHOLD>
    <EVENT-KIND>SWC</EVENT-KIND>
    <PRESTORAGE-FREEZE-FRAME>false</PRESTORAGE-FREEZE-FRAME>
</DIAGNOSTIC-EVENT>
```

Class	**DiagnosticConnectedIndicator**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticEvent			
Note	Description of indicators that are defined per DiagnosticEvent.			
Base	ARObject,Identifiable,MultilanguageReferrable,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
behavior	DiagnosticConnectedIndicatorBehaviorEnum	0..1	attr	Behavior of the linked indicator.

Attribute	Datatype	Mul.	Kind	Note
healingCycle	DiagnosticOperationCycle	1	ref	The deactivation of indicators per event is defined as healing of a diagnostic event. The operation cycle in which the warning indicator will be switched off is defined here.
indicator	DiagnosticIndicator	1	ref	Reference to the used indicator.

Table 6.2: DiagnosticConnectedIndicator

Enumeration	DiagnosticEventClearAllowedEnum
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticEvent
Note	Denotes whether clearing of events is allowed.
Literal	Description
always	The clearing is allowed unconditionally.
never	The clearing is never allowed.
requires Callback Execution	In case the clearing of a Diagnostic Event has to be allowed or prohibited through the SWC interface CallbackClearEventAllowed, the SWC has to indicate this by defining appropriate ServiceNeeds (i.e. DiagnosticEventNeeds).

Table 6.3: DiagnosticEventClearAllowedEnum

Enumeration	DiagnosticClearEventBehaviorEnum
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticEvent
Note	Possible behavior for clearing events.
Literal	Description
noStatusByte Change	The event status byte keeps unchanged.
onlyThis CycleAnd Readiness	The OperationCycle and readiness bits of the event status byte are reset.

Table 6.4: DiagnosticClearEventBehaviorEnum

Enumeration	DiagnosticEventKindEnum
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticEvent
Note	Applicability of the diagnostic event.
Literal	Description
bsw	The event is assigned to a BSW module.
swc	The event is assigned to a SWC.

Table 6.5: DiagnosticEventKindEnum

Enumeration	DiagnosticConnectedIndicatorBehaviorEnum
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticEvent
Note	Behavior of the indicator.
Literal	Description

blinkMode	The indicator blinks when the event has status FAILED.
blinkOrContinuousOnMode	The indicator is active and blinks when the event has status FAILED.
continuousOnMode	The indicator is active when the event has status FAILED.
fastFlashingMode	Flash Indicator Lamp should be set to "Fast Flash".
slowFlashingMode	Flash Indicator Lamp should be set to "Slow Flash".

Table 6.6: DiagnosticConnectedIndicatorBehaviorEnum

6.3 DiagnosticTroubleCode

DiagnosticTroubleCodes (i.e. the ECU external view on diagnostic events) are defined together with their properties and mapped to DiagnosticEvents using DiagnosticEventToTroubleCodeUdsMapping.

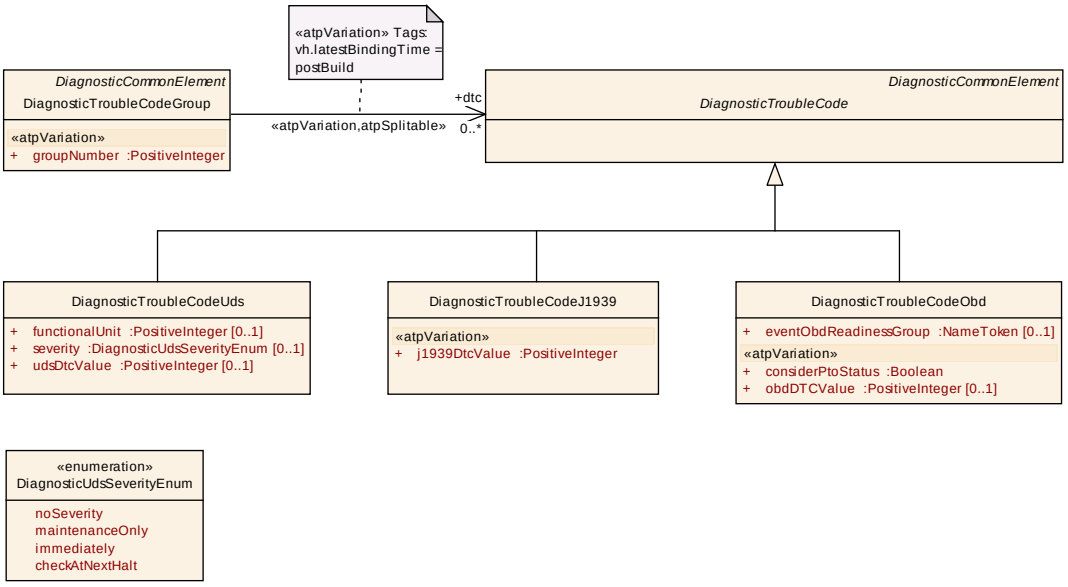

Figure 6.3: Modeling of DiagnosticTroubleCode

[TPS_DEXT_03012] Three kinds of DTCs ⌈ There are three kinds of DTCs represented as specializations of DiagnosticTroubleCode:

- non OBD relevant DTCs (DiagnosticTroubleCodeUds)
- OBD relevant DTCs (DiagnosticTroubleCodeObd)
- J1939 [17] relevant DTCs (DiagnosticTroubleCodeJ1939)

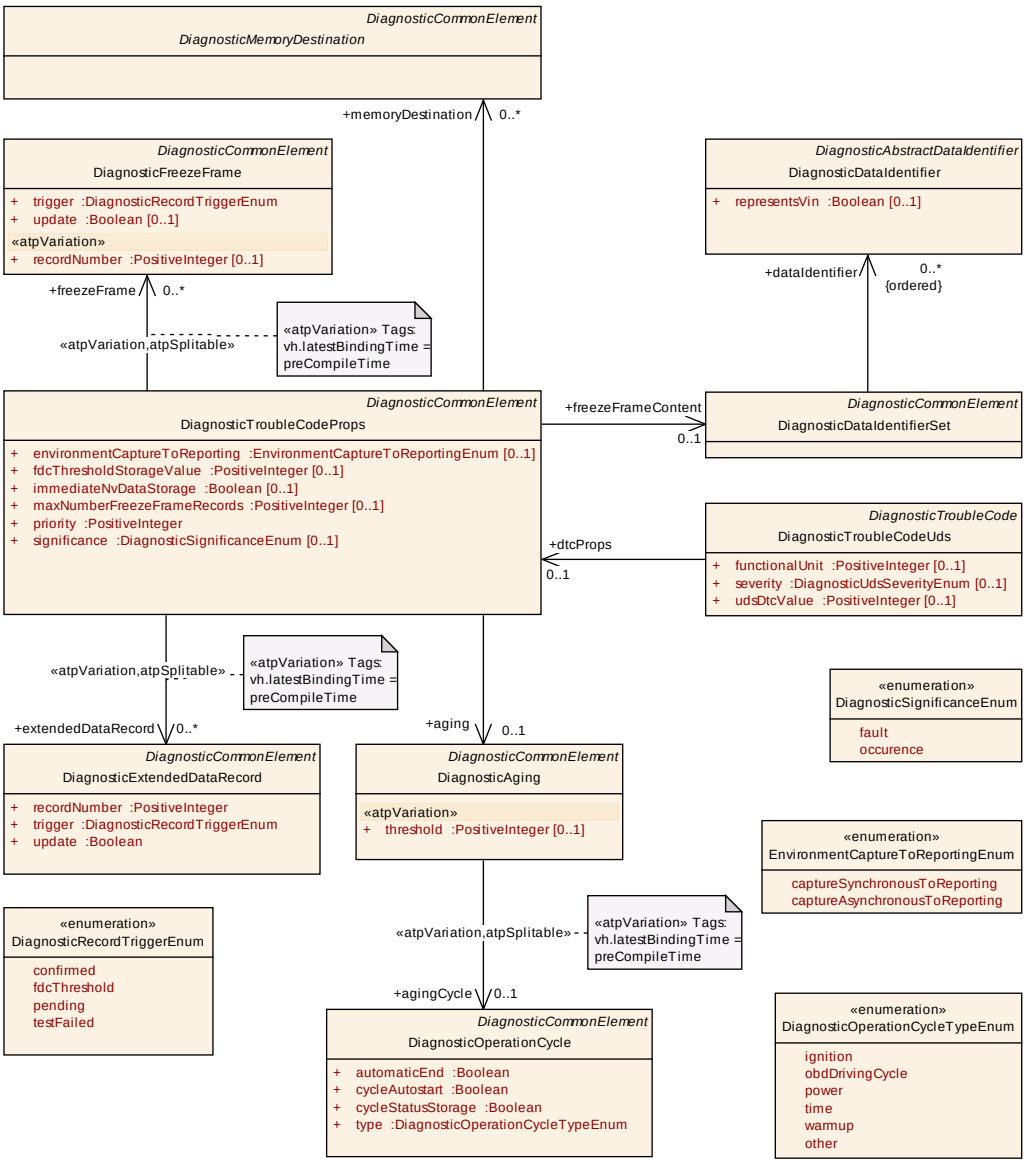

Figure 6.4: Modeling of `DiagnosticTroubleCodeUds`

Properties individual to such a DTC specialization are modeled as attributes of `DiagnosticTroubleCodeUds`, `DiagnosticTroubleCodeObd` and `DiagnosticTroubleCodeJ1939`, respectively. ⌋*(RS_DEXT_00024)*

[TPS_DEXT_03013] Common properties of a DTC ⌈ Properties that are often common for a group of `DiagnosticTroubleCodeUds` elements are modeled as attributes of `DiagnosticTroubleCodeProps`. ⌋*(RS_DEXT_00024)*

[constr_1349] Value of `udsDtcValue` shall be unique ⌈ The value of `udsDtcValue` shall be unique to any other DTC and DTC group value. ⌋*()*

Class	**DiagnosticTroubleCode (abstract)**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticTroubleCode			
Note	A diagnostic trouble code defines a unique identifier that is shown to the diagnostic tester.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
–	–	–	–	–

Table 6.7: DiagnosticTroubleCode

[TPS_DEXT_03014] Semantics of `DiagnosticTroubleCodeGroup` ⌈ The `DiagnosticTroubleCodeGroup` element is used to define groups of DTCs that belong together. Each `DiagnosticTroubleCodeGroup` has its own `groupNumber` value assigned. ⌋*(RS_DEXT_00024)*

Class	**DiagnosticTroubleCodeGroup**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticTroubleCode			
Note	The diagnostic trouble code group defines the DTCs belonging together and thereby forming a group.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
dtc	DiagnosticTroubleCode	*	ref	This represents the collection of DiagnosticTroubleCodes defined by this DiagnosticTroubleCodeGroup. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=dtc, variationPoint.shortLabel vh.latestBindingTime=postBuild
groupNumber	PositiveInteger	1	attr	This represents the base number of the DTC group. **Stereotypes:** atpVariation **Tags:** vh.latestBindingTime=preCompileTime

Table 6.8: DiagnosticTroubleCodeGroup

[constr_1350] Value of `DiagnosticTroubleCodeGroup.groupNumber` shall be unique ⌈ The value of `DiagnosticTroubleCodeGroup.groupNumber` shall be unique to any other DTC and DTC group value. ⌋*()*

[constr_1351] Value of `DiagnosticTroubleCodeGroup.groupNumber` ⌈ To be compliant to ISO, the value of `DiagnosticTroubleCodeGroup.groupNumber` shall be set as defined in ISO 14229-1 [15]. ⌋*()*

[TPS_DEXT_03000] ISO 14229-1 reserves values of `DiagnosticTroubleCodeGroup.groupNumber` ⌈ Any values other than those mentioned in [constr_1351] are reserved by ISO 14229-1 [15]. ⌋*(RS_DEXT_00024)*

[constr_1352] Existence of maxNumberFreezeFrameRecords vs. freezeFrame ⌈ If the attribute DiagnosticTroubleCodeProps.maxNumberFreezeFrameRecords exists than the attribute DiagnosticTroubleCodeProps.freezeFrame shall not exist or vice versa. ⌋*()*

[constr_1353] Applicability of [constr_1352] ⌈ [constr_1352] shall apply in the identical way (either one or the other attribute shall exist) for all DiagnosticTroubleCodeProps within the context of all DiagnosticContributionSets of category DIAGNOSTIC_ECU_EXTRACT that refer to the same EcuInstance. ⌋*()*

[constr_1354] Existence of attribute DiagnosticTroubleCodeProps.freezeFrameContent ⌈ If one of the attributes DiagnosticTroubleCodeProps.maxNumberFreezeFrameRecords or DiagnosticTroubleCodeProps.freezeFrame exists then the attribute DiagnosticTroubleCodeProps.freezeFrameContent shall exist. ⌋*()*

[TPS_DEXT_01064] Textually formulated content attached to DiagnosticTroubleCode ⌈ The definition of a DiagnosticTroubleCode also consists of textually formulated content that is formalized in structure but cannot be formalized in content.

The purpose of this content is to define e.g. an error text or the possible cause that relates to the specific DiagnosticTroubleCode. ⌋*(RS_DEXT_00024, RS_DEXT_00045)*

[TPS_DEXT_01065] Different approaches to provide semi-formal textual content attached to a DiagnosticTroubleCode ⌈ There are different approaches to provide semi-formal textual content attached to a DiagnosticTroubleCode:

- Textual description that has the character of descriptions of the DiagnosticTroubleCode shall be provided by means of the meta-class TraceableText, i.e. by means of introduction.trace.
- Textual description that characterizes the DiagnosticTroubleCode with respect to the *ODX long name* shall be provided by means of the attribute longName.

⌋*(RS_DEXT_00024, RS_DEXT_00045)*

For more details regarding the modeling of the semi-formal text please refer to Figure 4.3.

The usage of TraceableText and StructuredReq alone would not qualify as a semi-formal textual attachment. It is necessary to standardize the value of the category in order to get some level of semi-formal textual description.

[TPS_DEXT_01066] Standardized values of DiagnosticTroubleCode.introduction.trace ⌈ The following possible values of DiagnosticTroubleCode.introduction.trace are standardized by AUTOSAR:

- **DIAG_DTC_ERROR_TEXT**: this value shall be used to describe an error text.

- **DIAG_DTC_REP_ACT**: this value describes the associated repair for the corresponding DiagnosticTroubleCode.
- **DIAG_DTC_CUS_PER_SYMP**: this value describes the possible customer perception symptom for the corresponding DiagnosticTroubleCode.
- **DIAG_DTC_POSS_CAUSE**: This value describes the possible cause for the corresponding DiagnosticTroubleCode.

⌋(*RS_DEXT_00001*, *RS_DEXT_00024*, *RS_DEXT_00045*)

The following ARXML fragment exemplifies the usage of TraceableText along with the standardized values of the attribute category to attach semi-formal textual descriptions to a DiagnosticTroubleCodeUds.

Listing 6.2: Example for the definition of a semi-formal textual elements in the context of a DiagnosticTroubleCode

```
<DIAGNOSTIC-TROUBLE-CODE-UDS>
    <SHORT-NAME>ExampleDTC_0001</SHORT-NAME>
    <LONG-NAME>
        <L-4 L="EN">My little ODX long name</L-4>
    </LONG-NAME>
    <DESC>
        <L-2 L="EN">This DTC is a System Error DTC</L-2>
    </DESC>
    <INTRODUCTION>
        <TRACE>
            <SHORT-NAME>MyErrorText</SHORT-NAME>
            <CATEGORY>DIAG_DTC_ERROR_TEXT</CATEGORY>
            <P>
                <L-1 L="LA">Lorem ipsum dolor sit amet, consectetur
                    adipisicing elit</L-1>
            </P>
        </TRACE>
    </INTRODUCTION>
    <DTC-PROPS-REF DEST="DIAGNOSTIC-TROUBLE-CODE-PROPS">/AUTOSAR/
       UseCase_230/ExampleDTC_0001_Props</DTC-PROPS-REF>
    <FUNCTIONAL-UNIT>1</FUNCTIONAL-UNIT>
    <SEVERITY>CHECK-AT-NEXT-HALT</SEVERITY>
    <UDS-DTC-VALUE>0x000001</UDS-DTC-VALUE>
</DIAGNOSTIC-TROUBLE-CODE-UDS>
```

[constr_1376] Multiplicity of reference DiagnosticTroubleCodeProps.memoryDestination ⌈ For every given DiagnosticTroubleCodeProps, the reference in the role DiagnosticTroubleCodeProps.memoryDestination **shall not exceed** the upper multiplicity 2. [constr_1377] applies. ⌋()

[constr_1377] Existence of reference DiagnosticTroubleCodeProps.memoryDestination ⌈ The reference DiagnosticTroubleCodeProps.memoryDestination shall **only** have the upper multiplicity 2 **if one (and only one) of** the referenced DiagnosticTroubleCodeProps.memoryDestination is a DiagnosticMemoryDestinationMirror. ⌋()

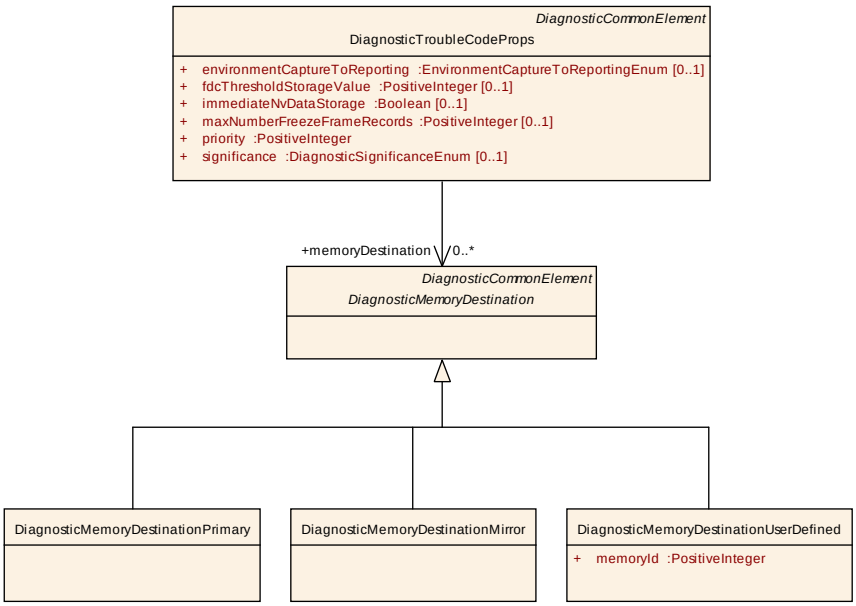

Figure 6.5: Modeling of `DiagnosticMemoryDestination`

[constr_1378] Value of `DiagnosticMemoryDestinationUserDefined.memoryId` ⌈ Within the scope of one `DiagnosticContributionSet`, no two (or more) `DiagnosticMemoryDestinationUserDefined`s shall exist that share the same value for attribute `DiagnosticMemoryDestinationUserDefined.memoryId` ⌋*()*

In other words, the value of the attribute `DiagnosticMemoryDestinationUserDefined.memoryId` shall be unique within any given `DiagnosticExtract`.

On top of that, it is necessary to make sure that only **one** *primary memory* and only **one** *mirror memory* is defined.

[constr_1379] Existence of `DiagnosticMemoryDestinationPrimary` ⌈ Within the scope of one `DiagnosticContributionSet` only one `DiagnosticMemoryDestinationPrimary` shall exist. ⌋*()*

[constr_1380] Existence of `DiagnosticMemoryDestinationMirror` ⌈ Within the scope of one `DiagnosticContributionSet` only one `DiagnosticMemoryDestinationMirror` shall exist. ⌋*()*

Class	**DiagnosticTroubleCodeUds**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticTroubleCode			
Note	This element is used to describe non OBD-relevant DTCs.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticTroubleCode,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
dtcProps	DiagnosticTroubleCodeProps	0..1	ref	Defined properties associated with the DemDTC.
functionalUnit	PositiveInteger	0..1	attr	This attribute specifies a 1-byte value which identifies the corresponding basic vehicle / system function which reports the DTC. This parameter is necessary for the report of severity information.

Attribute	Datatype	Mul.	Kind	Note
severity	DiagnosticUdsSeverityEnum	0..1	attr	DTC severity according to ISO 14229-1.
udsDtcValue	PositiveInteger	0..1	attr	Unique Diagnostic Trouble Code value for UDS.

Table 6.9: DiagnosticTroubleCodeUds

Class	DiagnosticTroubleCodeObd			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticTroubleCode			
Note	This element is used to define OBD-relevant DTCs.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticTroubleCode,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
considerPtoStatus	Boolean	1	attr	This attribute describes the affection of the event by the Dem PTO handling. True: the event is affected by the Dem PTO handling. False: the event is not affected by the Dem PTO handling. **Stereotypes:** atpVariation **Tags:** vh.latestBindingTime=preCompileTime
eventObdReadinessGroup	NameToken	0..1	attr	This attribute specifies the Event OBD Readiness group for PID $01 and PID $41 computation. This attribute is only applicable for emission-related ECUs.
obdDTCValue	PositiveInteger	0..1	attr	Unique Diagnostic Trouble Code value for OBD. **Stereotypes:** atpVariation **Tags:** vh.latestBindingTime=preCompileTime

Table 6.10: DiagnosticTroubleCodeObd

Class	DiagnosticTroubleCodeProps			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticTroubleCode			
Note	This element defines common Dtc properties that can be reused by different non OBD-relevant DTCs.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
aging	DiagnosticAging	0..1	ref	Reference to an aging algorithm in case that an aging/unlearning of the event is allowed.
environmentCaptureToReporting	EnvironmentCaptureToReportingEnum	0..1	attr	This attribute determines the point in time, when the data actually is captured.

Attribute	Datatype	Mul.	Kind	Note
extendedDataRecord	DiagnosticExtendedDataRecord	*	ref	Defines the links to an extended data class sampler. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=shortName, variationPoint.shortLabel vh.latestBindingTime=preCompileTime
fdcThresholdStorageValue	PositiveInteger	0..1	attr	Threshold to allocate an event memory entry and to capture the Freeze Frame. Unit: none (attribute represents a counter value).
freezeFrame	DiagnosticFreezeFrame	*	ref	Define the links to a freeze frame class sampler. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=shortName, variationPoint.shortLabel vh.latestBindingTime=preCompileTime
freezeFrameContent	DiagnosticDataIdentifierSet	0..1	ref	This represents the content of the a set of DiagnosticFreezeFrames.
immediateNvDataStorage	Boolean	0..1	attr	Switch to enable immediate storage triggering of an according event memory entry persistently to NVRAM. true: immediate non-volatile storage triggering enabled false: immediate non-volatile storage triggering disabled
maxNumberFreezeFrameRecords	PositiveInteger	0..1	attr	This attribute defines the number of according freeze frame records, which can maximal be stored for this event. Therefore all these freeze frame records have the same freeze frame class.
memoryDestination	DiagnosticMemoryDestination	*	ref	The event destination assigns events to none, one or multiple origins.
priority	PositiveInteger	1	attr	Priority of the event, in view of full event buffer. A lower value means higher priority.
significance	DiagnosticSignificanceEnum	0..1	attr	Significance of the event, which indicates additional information concerning fault classification and resolution.

Table 6.11: DiagnosticTroubleCodeProps

Class	DiagnosticMemoryDestination (abstract)			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticTroubleCode			
Note	This abstract meta-class represents a possible memory destination for a diagnostic event.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
–	–	–	–	–

Table 6.12: DiagnosticMemoryDestination

Class	DiagnosticMemoryDestinationPrimary			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticTroubleCode			
Note	This represents a primary memory for a diagnostic event.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticMemoryDestination,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
–	–	–	–	–

Table 6.13: DiagnosticMemoryDestinationPrimary

Class	DiagnosticMemoryDestinationMirror			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticTroubleCode			
Note	This represents a mirror memory for a diagnostic event.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticMemoryDestination,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
–	–	–	–	–

Table 6.14: DiagnosticMemoryDestinationMirror

Class	DiagnosticMemoryDestinationUserDefined			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticTroubleCode			
Note	This represents a user-defined memory for a diagnostic event.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticMemoryDestination,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
memoryId	PositiveInteger	1	attr	This represents the identifier of the user-defined memory.

Table 6.15: DiagnosticMemoryDestinationUserDefined

Enumeration	DiagnosticSignificanceEnum
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticTroubleCode
Note	Significance level of a diagnostic event.
Literal	**Description**
fault	Failure, which affects the component/ECU itself.
occurence	Issue, which indicates additional information concerning insufficient system behavior.

Table 6.16: DiagnosticSignificanceEnum

Enumeration	DiagnosticUdsSeverityEnum
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticTroubleCode
Note	Severity types for a DTC according to ISO 14229-1.

Literal	*Description*
checkAtNext Halt	Check at next halt.
immediately	Check immediately.
maintenance Only	Maintenance required.
noSeverity	No severity information available.

Table 6.17: DiagnosticUdsSeverityEnum

Class	**DiagnosticDataIdentifierSet**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticTroubleCode			
Note	This represents the ability to define a list of DiagnosticDataIdentifiers that can be reused in different contexts.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommon Element,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
dataId entifier (ordered)	DiagnosticDataI dentifier	*	ref	Reference to an orderd list of Data Identifiers.

Table 6.18: DiagnosticDataIdentifierSet

6.4 DiagnosticExtendedDataRecord

[TPS_DEXT_03008] Semantics of `DiagnosticExtendedDataRecord` ⌈ A `DiagnosticExtendedDataRecord` contains `DiagnosticDataElement`s that are ordered by the `bitOffset`. ⌋*(RS_DEXT_00032)*

[constr_1355] Value of `recordNumber` ⌈ To be compliant to ISO, the value of `recordNumber` shall be set in the interval as defined in ISO 14229-1 [15]. ⌋*()*

[constr_1356] Value of `recordNumber` shall be unique ⌈ The value of `recordNumber` shall be unique among all `DiagnosticExtendedDataRecord`s in the context of the enclosing `DiagnosticContributionSet`. ⌋*()*

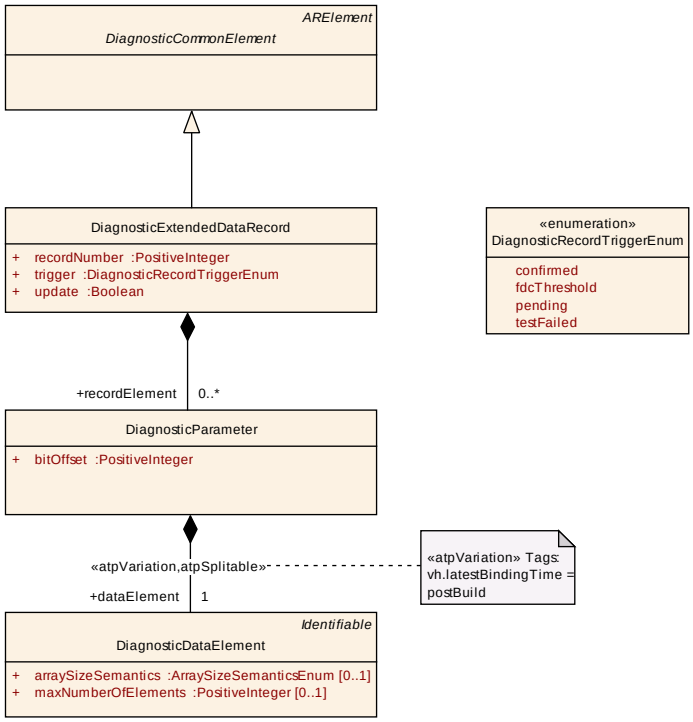

Figure 6.6: Modeling of DiagnosticExtendedDataRecord

Class	**DiagnosticExtendedDataRecord**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticExtendedDataRecord			
Note	Description of an extended data record.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
recordElement	DiagnosticParameter	*	aggr	Defined DataElements in the extended record element.
recordNumber	PositiveInteger	1	attr	This attribute specifies an unique identifier for an extended data record.
trigger	DiagnosticRecordTriggerEnum	1	attr	This attribute specifies the primary trigger to allocate an event memory entry.
update	Boolean	1	attr	This attribute defines when an extended data record is captured. True: This extended data record is captured every time. False: This extended data record is only captured for new event memory entries.

Table 6.19: DiagnosticExtendedDataRecord

Enumeration	**DiagnosticRecordTriggerEnum**
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticFreezeFrame
Note	Triggers to allocate an event memory entry.
Literal	***Description***
confirmed	capture on "Confirmed"
fdcThreshold	capture on "FDC Threshold"

pending	capture on "Pending"
testFailed	capture on "Test Failed"

Table 6.20: DiagnosticRecordTriggerEnum

6.5 DiagnosticFreezeFrame

[TPS_DEXT_03009] Semantics of `DiagnosticFreezeFrame` ⌈ A `DiagnosticFreezeFrame` needs an ordered list of references to `DiagnosticDataIdentifier`s. However, this reference is not modeled directly but in the context of meta-class `DiagnosticTroubleCodeProps`. ⌋*(RS_DEXT_00033)*

For more details, please refer to Figure 6.3.

[constr_1357] Value of `recordNumber` ⌈ To be compliant to ISO, the value of `recordNumber` shall be set in the interval as defined in ISO 14229-1 [15]. ⌋*()*

[constr_1358] Value of `recordNumber` shall be unique ⌈ The value of `recordNumber` shall be unique among all `DiagnosticFreezeFrame`s in the context of the enclosing `DiagnosticContributionSet`. ⌋*()*

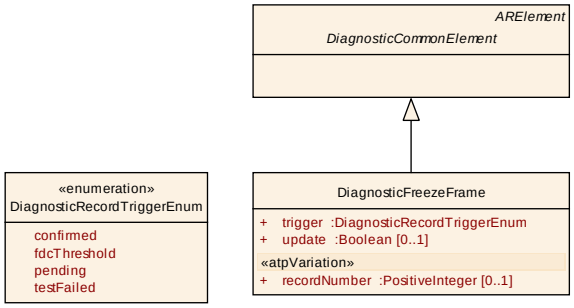

Figure 6.7: Modeling of DiagnosticFreezeFrame

Class	**DiagnosticFreezeFrame**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticFreezeFrame			
Note	This element describes combinations of DIDs for a non OBD relevant freeze frame.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
recordNumber	PositiveInteger	0..1	attr	This attribute defines a record number for a freeze frame record. **Stereotypes:** atpVariation **Tags:** vh.latestBindingTime=preCompileTime
trigger	DiagnosticRecordTriggerEnum	1	attr	This attribute defines the primary trigger to allocate an event memory entry.

Attribute	*Datatype*	*Mul.*	*Kind*	*Note*
update	Boolean	0..1	attr	This attribute defines the approach when the freeze frame record is stored/updated. True: FreezeFrame record is captured every time. False: FreezeFrame record is only captured for new event memory entries.

Table 6.21: DiagnosticFreezeFrame

6.6 DiagnosticCondition

[TPS_DEXT_03010] Combination of DiagnosticConditions to DiagnosticConditionGroups ⌈ DiagnosticConditions are combined to DiagnosticConditionGroups and define a certain number of checks (e.g. correct voltage range) before the event report is accepted or the event gets qualified. ⌋*(RS_DEXT_00027, RS_DEXT_00028, RS_DEXT_00030, RS_DEXT_00031)*

[TPS_DEXT_03001] Different types of conditions ⌈ There are two different types of conditions: DiagnosticEnableConditions and DiagnosticStorageCondition:

- As long as the DiagnosticEnableCondition is not fulfilled, the event reports are not valid and therefore will not be accepted.
- As long as the DiagnosticStorageCondition is not fulfilled, the event is not stored in the event memory.

⌋*(RS_DEXT_00027)*

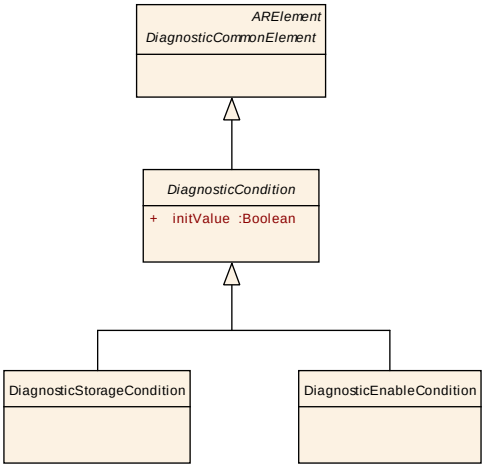

Figure 6.8: Modeling of DiagnosticCondition

Class	DiagnosticCondition (abstract)			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticCondition			
Note	Abstract element for StorageConditions and EnableConditions.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommon Element,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
initValue	Boolean	1	attr	Defines the initial status for enable or disable of acceptance/storage of event reports of a diagnostic event. The value is the initialization after power up (before this condition is reported the first time). true: acceptance/storage of a diagnostic event enabled false: acceptance/storage of a diagnostic event disabled

Table 6.22: DiagnosticCondition

Class	DiagnosticEnableCondition			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticCondition			
Note	Specification of an enable condition.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Diagnostic Condition,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
–	–	–	–	–

Table 6.23: DiagnosticEnableCondition

Class	DiagnosticStorageCondition			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticCondition			
Note	Specification of a storage condition.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Diagnostic Condition,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
–	–	–	–	–

Table 6.24: DiagnosticStorageCondition

6.7 DiagnosticConditionGroup

[TPS_DEXT_01084] Semantics of `DiagnosticConditionGroup`s ⌈ `DiagnosticConditionGroup`s are used to collect `DiagnosticCondition`s that in turn are assigned to `DiagnosticEvent`s. ⌋*(RS_DEXT_00023, RS_DEXT_00028, RS_DEXT_00029)*

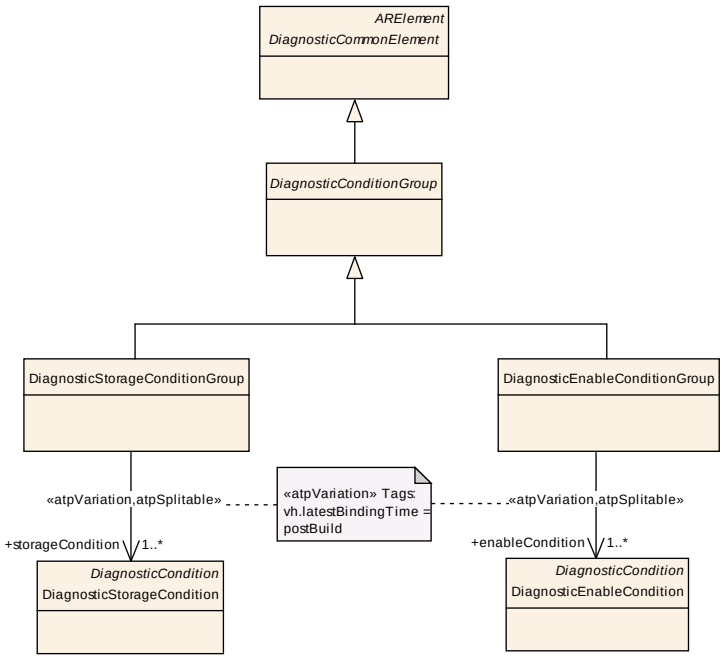

Figure 6.9: Modelling of DiagnosticConditionGroup

Class	**DiagnosticConditionGroup (abstract)**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticConditionGroup			
Note	Abstract element for StorageConditionGroups and EnableConditionGroups.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommon Element,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
–	–	–	–	–

Table 6.25: DiagnosticConditionGroup

Class	**DiagnosticEnableConditionGroup**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticConditionGroup			
Note	Enable condition group which includes one or several enable conditions.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Diagnostic ConditionGroup,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
enableCon dition	DiagnosticEnabl eCondition	1..*	ref	Reference to enableConditions that are part of the EnableConditionGroup. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=enableCondition, variation Point.shortLabel vh.latestBindingTime=postBuild

Table 6.26: DiagnosticEnableConditionGroup

Class	**DiagnosticStorageConditionGroup**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticConditionGroup			
Note	Storage condition group which includes one or several storage conditions.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticConditionGroup,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
storageCondition	DiagnosticStorageCondition	1..*	ref	Reference to storageConditions that are part of the StorageConditionGroup. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=storageCondition, variationPoint.shortLabel vh.latestBindingTime=postBuild

Table 6.27: DiagnosticStorageConditionGroup

6.8 DiagnosticMapping

The mapping concept of the `DiagnosticExtract` template has been designed to support the decentralized and independent definition of diagnostic requirements that can be linked together at a late point during the development process.

It also supports the use of mapping contributions collected from various sources in order to reduce manual mapping work by the ECU integrator.

[TPS_DEXT_03002] Two kind of mappings ⌈ For diagnostic event handling, there are two kind of mappings:

- Mapping between a `DiagnosticEvent` and another diagnostic definition.
- Mapping between a `DiagnosticEvent` and a SWC service port.

⌋*(RS_DEXT_00023, RS_DEXT_00052)*

Figure 6.10 gives an overview on the different types of mappings available for diagnostic event handling.

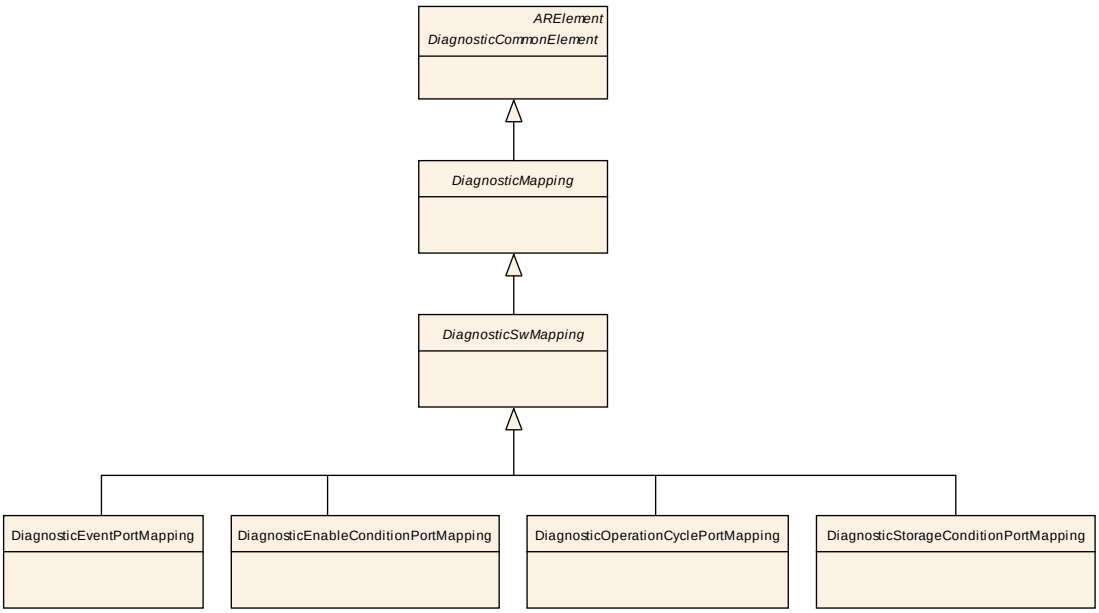

Figure 6.10: Modelling of DiagnosticMapping

Class	**DiagnosticMapping (abstract)**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticMapping			
Note	Abstract element for different kinds of diagnostic mappings.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
–	–	–	–	–

Table 6.28: DiagnosticMapping

6.8.1 DiagnosticEvent to DtcUds Mapping

[TPS_DEXT_03003] Semantics of `DiagnosticEventToTroubleCodeUdsMapping` ⌈ The `DiagnosticEventToTroubleCodeUdsMapping` is used to assign one (1:1) or multiple (n:1) `DiagnosticEvent`s to a `DiagnosticTroubleCodeUds`.

In case of n:1, multiple instances of `DiagnosticEventToTroubleCodeUdsMapping` with the same reference of role `troubleCodeUds` but different references of role `diagnosticEvent` have to be defined. ⌋*(RS_DEXT_00023, RS_DEXT_00024)*

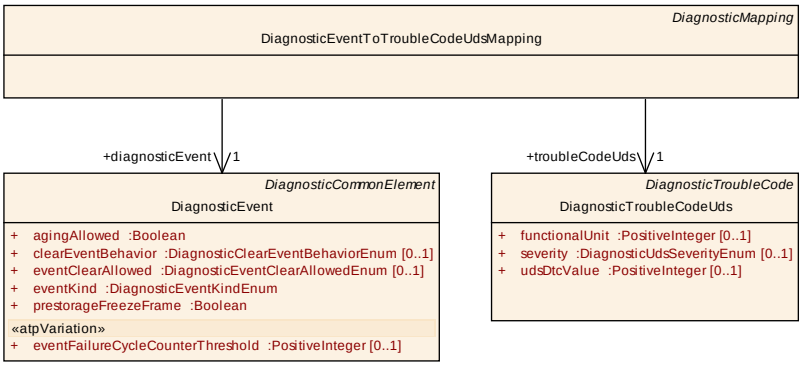

Figure 6.11: DiagnosticEventToDtcUdsMapping

Class	**DiagnosticEventToTroubleCodeUdsMapping**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticMapping			
Note	Defines which UDS Diagnostic Trouble Code is applicable for a DiagnosticEvent.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticMapping,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
diagnosticEvent	DiagnosticEvent	1	ref	Reference to a DiagnosticEvent to which a UDS Diagnostic Trouble Code is assigned.
troubleCodeUds	DiagnosticTroubleCodeUds	1	ref	Reference to an UDS Diagnostic Trouble Code assigned to a DiagnosticEvent.

Table 6.29: DiagnosticEventToTroubleCodeUdsMapping

6.8.2 DiagnosticEvent to DiagnosticOperationCycle Mapping

[TPS_DEXT_01086] Reference to `DiagnosticOperationCycle` ⌈ A `DiagnosticEvent` needs to be assigned to exactly one `DiagnosticOperationCycle`. ⌋*(RS_DEXT_00024, RS_DEXT_00054)*

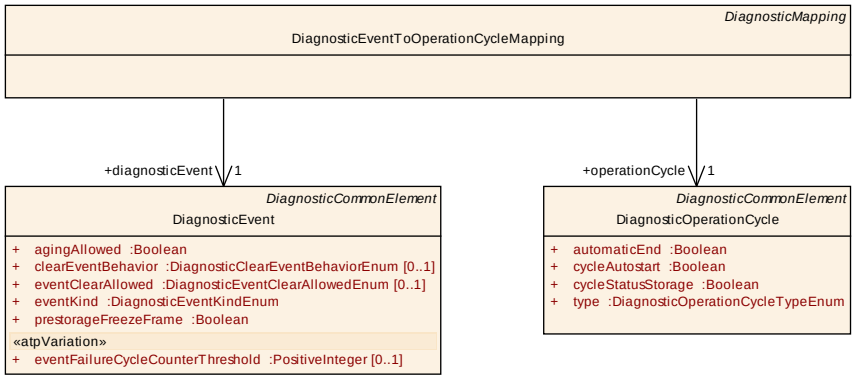

Figure 6.12: DiagnosticEventToOperationCycleMapping

Class	DiagnosticEventToOperationCycleMapping			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticMapping			
Note	Defines which OperationCycle is applicable for a DiagnosticEvent.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticMapping,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
diagnosticEvent	DiagnosticEvent	1	ref	Reference to a DiagnosticEvent to which an OperationCycle is assigned.
operationCycle	DiagnosticOperationCycle	1	ref	Reference to an OperationCycle assigned to a DiagnosticEvent.

Table 6.30: DiagnosticEventToOperationCycleMapping

6.8.3 DiagnosticEvent to DebounceAlgorithm Mapping

[TPS_DEXT_03004] `DiagnosticEvent` and `DiagnosticDebounceAlgorithmProps` ⌈ If a `DiagnosticEvent` has to be debounced, it must be mapped to the appropriate `DiagnosticDebounceAlgorithmProps`. ⌋*(RS_DEXT_00023, RS_DEXT_00053)*

[TPS_DEXT_03005] Existence of `DiagnosticEventToDebounceAlgorithmMapping` ⌈ The `DiagnosticEventToDebounceAlgorithmMapping` shall not be created if the `DiagnosticEvent` is not debounced. ⌋*(RS_DEXT_00023, RS_DEXT_00053)*

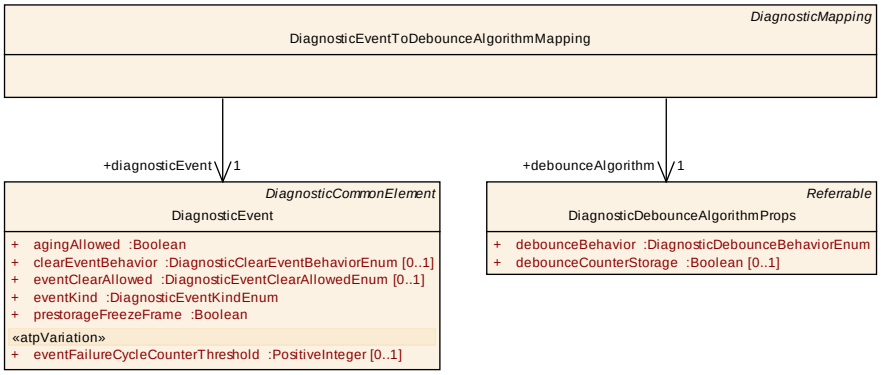

Figure 6.13: DiagnosticEventToDebounceAlgorithmMapping

Class	DiagnosticEventToDebounceAlgorithmMapping			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticMapping			
Note	Defines which Debounce Algorithm is applicable for a DiagnosticEvent.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticMapping,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
debounceAlgorithm	DiagnosticDebounceAlgorithmProps	1	ref	Reference to a DebounceAlgorithm assigned to a DiagnosticEvent.

Attribute	Datatype	Mul.	Kind	Note
diagnostic Event	DiagnosticEvent	1	ref	Reference to a DiagnosticEvent to which a DebounceAlgorithm is assigned.

Table 6.31: DiagnosticEventToDebounceAlgorithmMapping

Class	DiagnosticDebounceAlgorithmProps			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticDebouncingAlgorithm			
Note	Defines properties for the debounce algorithm class.			
Base	ARObject,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
debounce Algorithm	DiagEventDebounceAlgorithm	1	aggr	This represents the actual debounce algorithm.
debounce Behavior	DiagnosticDebounceBehaviorEnum	1	attr	This attribute defines how the event debounce algorithm will behave, if a related enable condition is not fulfilled or ControlDTCSetting of the related event is disabled.
debounce CounterStorage	Boolean	0..1	attr	Switch to store the debounce counter value non-volatile or not. true: debounce counter value shall be stored non-volatile false: debounce counter value is volatile

Table 6.32: DiagnosticDebounceAlgorithmProps

The details regarding the formalization of debouncing behavior are depicted in Figure 6.14.

In particular, `DiagnosticCommonProps` aggregates `DiagnosticDebounceAlgorithmProps` in the role `debounceAlgorithmProps`. The `DiagnosticDebounceAlgorithmProps` itself does not actually represent the debouncing algorithm but provides attributes relevant for the actual debouncing algorithm.

[TPS_DEXT_01048] Actual algorithm for the diagnostic event debouncing ⌈ The actual algorithm for the debouncing is represented by subclasses of `DiagEventDebounceAlgorithm` aggregated in the role `DiagnosticDebounceAlgorithmProps.debounceAlgorithm`. ⌋*(RS_DEXT_00023, RS_DEXT_00053)*

In other words, the debouncing of diagnostic events can be formulated in two ways:

- The `DiagEventDebounceCounterBased` represents the ability to implement a counter-based debouncing.
- The `DiagEventDebounceTimeBased` represents the ability to implement a time-based debouncing.

[constr_1359] Existence of attribute `DiagnosticDebounceAlgorithmProps.debounceCounterStorage` ⌈ The attribute `DiagnosticDebounceAlgorithmProps.debounceCounterStorage` shall only exist if the aggregation `DiagnosticDebounceAlgorithmProps.debounceAlgorithm` actually aggregates a `DiagEventDebounceCounterBased` ⌋*()*

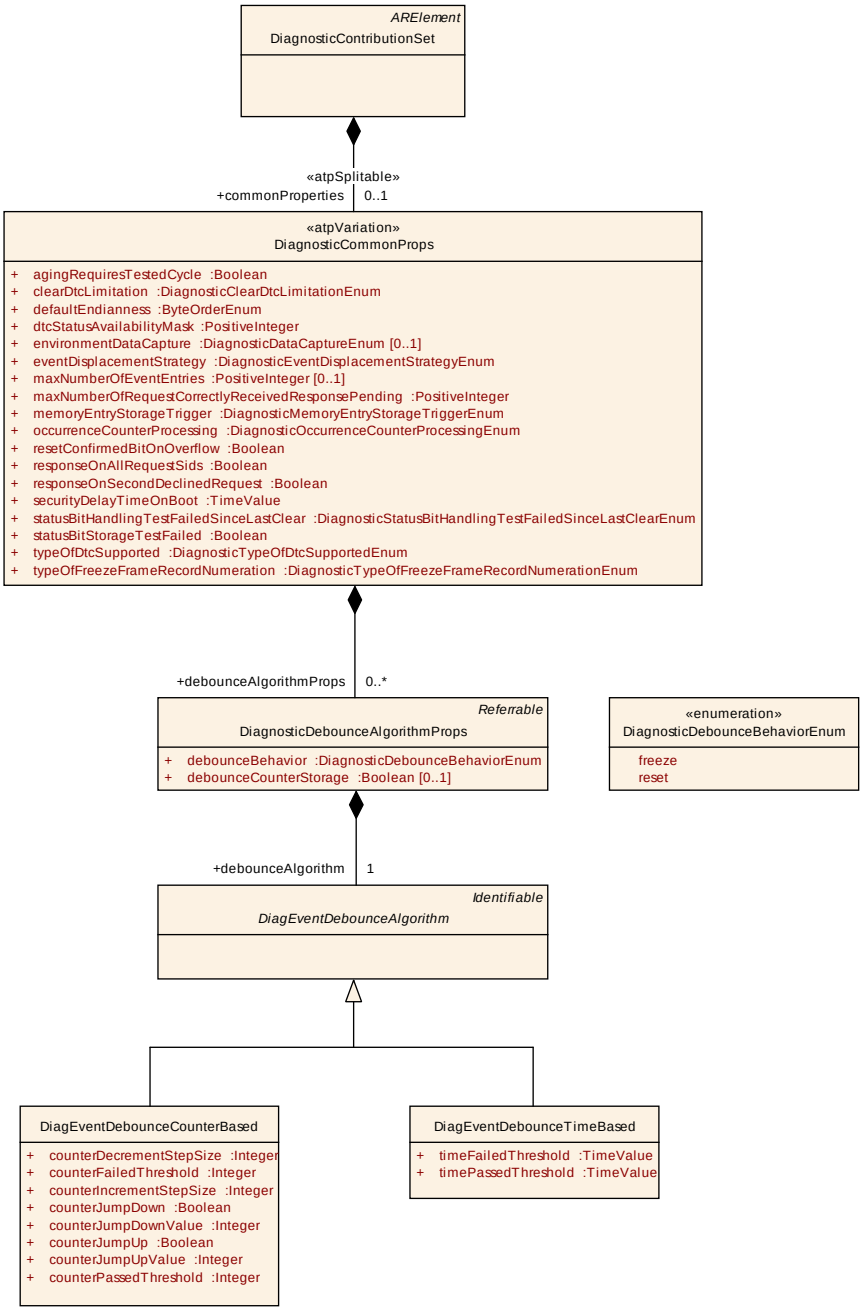

Figure 6.14: Details of `DiagnosticDebounceAlgorithmProps`

[constr_1360] Usage of `DiagEventDebounceMonitorInternal` is not supported in the context of `DiagnosticDebounceAlgorithmProps` ⌈ The usage of the meta-class `DiagEventDebounceMonitorInternal` for the aggregation in the role `DiagnosticDebounceAlgorithmProps.debounceAlgorithm` is not permitted. ⌋*()*

For clarification with respect to [constr_1360], `DiagEventDebounceMonitorInternal` is used in the context of formulating the `DiagnosticEventNeeds`, but its usage in the context of the `DiagnosticExtract` is not foreseen.

Class	**DiagEventDebounceCounterBased**			
Package	M2::AUTOSARTemplates::CommonStructure::ServiceNeeds			
Note	This meta-class represents the ability to indicate that the counter-based debounce algorithm shall be used by the DEM for this diagnostic monitor. This is related to set the ECUC choice container DemDebounceAlgorithmClass to DemDebounceCounterBased.			
Base	ARObject,DiagEventDebounceAlgorithm,Identifiable,MultilanguageReferrable,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
counterDecrementStepSize	Integer	1	attr	This value shall be taken to decrement the internal debounce counter.
counterFailedThreshold	Integer	1	attr	This value defines the event-specific limit that indicates the "failed" counter status.
counterIncrementStepSize	Integer	1	attr	This value shall be taken to increment the internal debounce counter.
counterJumpDown	Boolean	1	attr	This value activates or deactivates the counter jump-down behavior.
counterJumpDownValue	Integer	1	attr	This value represents the initial value of the internal debounce counter if the counting direction changes from incrementing to decrementing.
counterJumpUp	Boolean	1	attr	This value activates or deactivates the counter jump-up behavior.
counterJumpUpValue	Integer	1	attr	This value represents the initial value of the internal debounce counter if the counting direction changes from decrementing to incrementing.
counterPassedThreshold	Integer	1	attr	This value defines the event-specific limit that indicates the "passed" counter status.

Table 6.33: DiagEventDebounceCounterBased

Class	**DiagEventDebounceTimeBased**			
Package	M2::AUTOSARTemplates::CommonStructure::ServiceNeeds			
Note	This meta-class represents the ability to indicate that the time-based pre-debounce algorithm shall be used by the DEM for this diagnostic monitor. This is related to set the ECUC choice container DemDebounceAlgorithmClass to DemDebounceTimeBase.			
Base	ARObject,DiagEventDebounceAlgorithm,Identifiable,MultilanguageReferrable,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
timeFailedThreshold	TimeValue	1	attr	This value represents the event-specific delay indicating the "failed" status.
timePassedThreshold	TimeValue	1	attr	This value represents the event-specific delay indicating the "passed" status.

Table 6.34: DiagEventDebounceTimeBased

Enumeration	**DiagnosticDebounceBehaviorEnum**
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticDebouncing Algorithm
Note	Event debounce algorithm behavior options.
Literal	***Description***
freeze	The event debounce counter will be frozen with the current value and will not change while a related enable condition is not fulfilled or ControlDTCSetting of the related event is disabled. After all related enable conditions are fulfilled and ControlDTCSetting of the related event is enabled again, the event qualification will continue with the next report of the event (i.e. SetEventStatus).
reset	The event debounce counter will be reset to initial value if a related enable condition is not fulfilled or ControlDTCSetting of the related event is disabled. The qualification of the event will be restarted with the next valid event report.

Table 6.35: DiagnosticDebounceBehaviorEnum

6.8.4 DiagnosticEvent to EnableConditionGroup Mapping

[TPS_DEXT_03015] `EnableConditions` have to be put into a `DiagnosticEnableConditionGroup` ⌈ `EnableConditions` that are assigned to a `DiagnosticEvent` have to be put into a `DiagnosticEnableConditionGroup` since only a group of `EnableConditions` can be mapped to a `DiagnosticEvent`. ⌋*(RS_DEXT_00023, RS_DEXT_00026, RS_DEXT_00028)*

[constr_1361] Number of `DiagnosticEventToEnableConditionGroupMapping` elements per `DiagnosticEvent` ⌈ The mapping element `DiagnosticEventToEnableConditionGroupMapping` shall be created no more than once per `DiagnosticEvent`.

If several `DiagnosticEventToEnableConditionGroupMapping` elements referring the same `DiagnosticEvent` are defined, then the `Enable Condition Group` mapping shall be regarded as defective. ⌋*()*

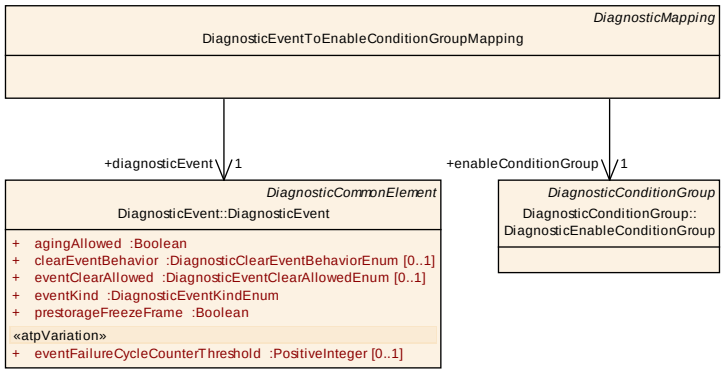

Figure 6.15: DiagnosticEventToEnableConditionGroupMapping

Class	**DiagnosticEventToEnableConditionGroupMapping**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticMapping			
Note	Defines which EnableConditionGroup is applicable for a DiagnosticEvent.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticMapping,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
diagnosticEvent	DiagnosticEvent	1	ref	Reference to a DiagnosticEvent to which an EnableConditionGroup is assigned.
enableConditionGroup	DiagnosticEnableConditionGroup	1	ref	Reference to an EnableConditionGroup assigned to a DiagnosticEvent.

Table 6.36: DiagnosticEventToEnableConditionGroupMapping

6.8.5 DiagnosticEvent to StorageConditionGroup Mapping

[TPS_DEXT_03016] `StorageConditions` have to be put into a `DiagnosticStorageConditionGroup` ⌈ `StorageConditions` that are assigned to a `DiagnosticEvent` have to be put into a `DiagnosticStorageConditionGroup` since only a group of `StorageConditions` can be mapped to a `DiagnosticEvent`. ⌋*(RS_DEXT_00023, RS_DEXT_00027, RS_DEXT_00029)*

[constr_1362] Number of `DiagnosticEventToStorageConditionGroupMapping` elements per `DiagnosticEvent` ⌈ The mapping element `DiagnosticEventToStorageConditionGroupMapping` shall be created no more than once or once per `DiagnosticEvent`.

If several `DiagnosticEventToStorageConditionGroupMapping` elements referring the same `DiagnosticEvent` are defined, then the `Storage Condition Group` mapping shall be regarded as defective. ⌋*()*

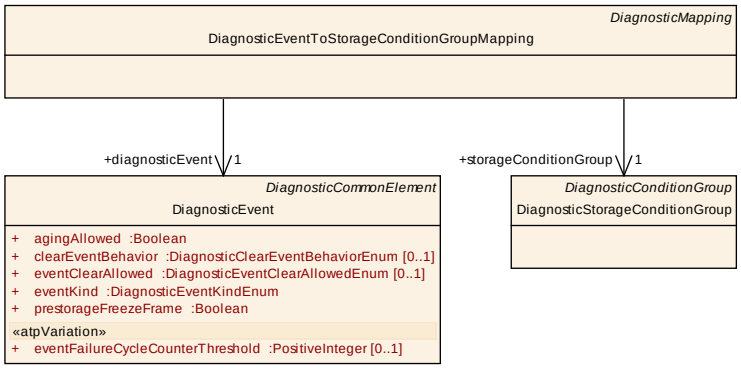

Figure 6.16: DiagnosticEventToStorageConditionGroupMapping

[TPS_DEXT_03006] Values of the individual `DiagnosticStorageConditions` ⌈ The values of the individual `DiagnosticStorageConditions` need to be algorithmically evaluated in order to find out whether or not the storage of the `DiagnosticEvent` is permitted.

The algorithm that is supposed to be implemented for this purpose is documented in [SWS_Dem_00459]. ⌋(*RS_DEXT_00027*)

Class	**DiagnosticEventToStorageConditionGroupMapping**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticMapping			
Note	Defines which StorageConditionGroup is applicable for a DiagnosticEvent.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticMapping,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
diagnosticEvent	DiagnosticEvent	1	ref	Reference to a DiagnosticEvent to which a StorageConditionGroup is assigned.
storageConditionGroup	DiagnosticStorageConditionGroup	1	ref	Reference to a StorageConditionGroup assigned to a DiagnosticEvent.

Table 6.37: DiagnosticEventToStorageConditionGroupMapping

6.8.6 DiagnosticEvent to Port Mapping

[TPS_DEXT_03007] Semantics of `DiagnosticEventPortMapping` ⌈ A `DiagnosticEventPortMapping` defines which `SwcServiceDependency`s of a `AtomicSwComponentType` or `BswServiceDependency` of a `BswModuleDescription` have to be connected to which `DiagnosticEvent`.

This is realized by defining a `DiagnosticEventPortMapping` referencing a `DiagnosticEvent` and (using `«instanceRef»` an instance of `SwcServiceDependency` (or `BswServiceDependency`). ⌋(*RS_DEXT_00023, RS_DEXT_00052*)

If such an instance is not yet available, an ordinary reference to `SwcServiceDependency` can be given alternatively (i.e. without specifying a certain instance).

In this way, the ECU integrator is able to directly derive the actual mapping between SWC (or BSW) service ports and the ports of the `Service Components` during ECU configuration.

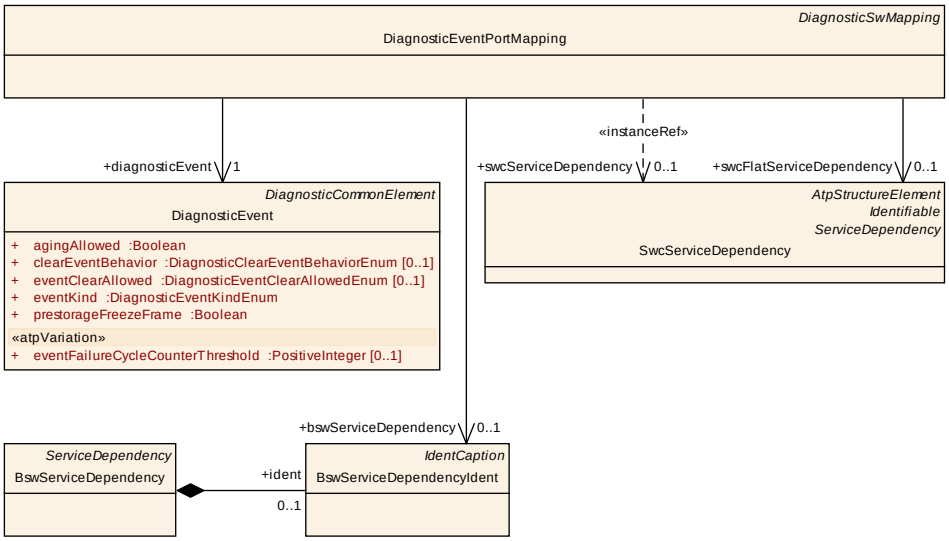

Figure 6.17: DiagnosticEventPortMapping

Class	**DiagnosticEventPortMapping**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticMapping			
Note	Defines to which SWC service ports with DiagnosticEventNeeds the DiagnosticEvent is mapped.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticMapping,DiagnosticSwMapping,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
bswServiceDependency	BswServiceDependencyIdent	0..1	ref	Reference to a BswServiceDependency that links ServiceNeeds to BswModuleEntries.
diagnosticEvent	DiagnosticEvent	1	ref	Reference to the DiagnosticEvent that is assigned to SWC service ports with DiagnosticEventNeeds.
swcFlatServiceDependency	SwcServiceDependency	0..1	ref	Reference to a SwcServiceDependencyType that links ServiceNeeds to SWC service ports.
swcServiceDependency	SwcServiceDependency	0..1	iref	Instance reference to a SwcServiceDependency that links ServiceNeeds to SWC service ports.

Table 6.38: DiagnosticEventPortMapping

6.8.7 DiagnosticOperationCycle to Port Mapping

[TPS_DEXT_03017] Semantics of `DiagnosticOperationCyclePortMapping` ⌈
A `DiagnosticOperationCyclePortMapping` defines which SWC service port(s) have to be connected to which `DiagnosticOperationCycle`.

This is realized by defining a `DiagnosticOperationCyclePortMapping` referencing a `DiagnosticOperationCycle` and an instance of `SwcServiceDependency`.
⌋*(RS_DEXT_00052, RS_DEXT_00053)*

If such an instance is not yet available, an ordinary reference to SwcServiceDependency can be given alternatively (i.e. without specifying a certain instance).

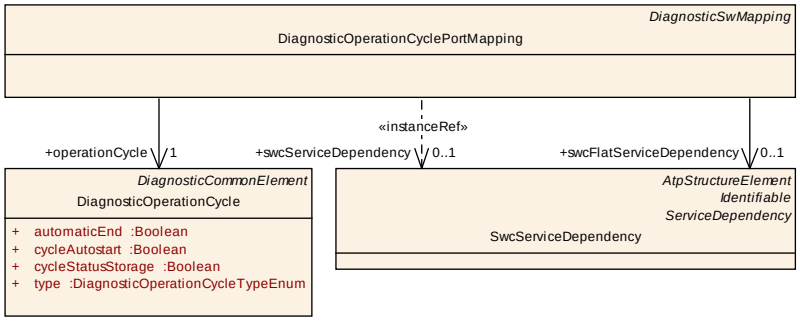

Figure 6.18: DiagnosticOperationCyclePortMapping

Class	**DiagnosticOperationCyclePortMapping**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticMapping			
Note	Defines to which SWC service ports with DiagnosticOperationCycleNeeds the DiagnosticOperationCycle is mapped.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticMapping,DiagnosticSwMapping,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
operationCycle	DiagnosticOperationCycle	1	ref	Reference to the DiagnosticOperationCycle that is assigned to SWC service ports with DiagnosticOperationCycleNeeds.
swcFlatServiceDependency	SwcServiceDependency	0..1	ref	Reference to a SwcServiceDependencyType that links ServiceNeeds to SWC service ports.
swcServiceDependency	SwcServiceDependency	0..1	iref	Instance reference to a SwcServiceDependency that links ServiceNeeds to SWC service ports.

Table 6.39: DiagnosticOperationCyclePortMapping

6.8.8 DiagnosticEnableCondition to Port Mapping

[TPS_DEXT_03018] Semantics of DiagnosticEnableConditionPortMapping
⌈ A DiagnosticEnableConditionPortMapping defines which SWC service port(s) have to be connected to which DiagnosticEnableCondition. This is realized by defining a DiagnosticEnableConditionPortMapping referencing a DiagnosticEnableCondition and an instance of SwcServiceDependency. ⌋*(RS_DEXT_00026, RS_DEXT_00052)*

If such an instance is not yet available, an ordinary reference to SwcServiceDependency can be given alternatively (i.e. without specifying a certain instance).

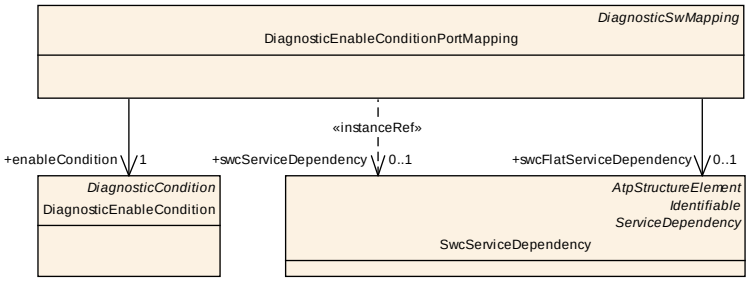

Figure 6.19: DiagnosticEnableConditionPortMapping

Class	**DiagnosticEnableConditionPortMapping**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticMapping			
Note	Defines to which SWC service ports with DiagnosticEnableConditionNeeds the DiagnosticEnableCondition is mapped.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticMapping,DiagnosticSwMapping,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
enableCondition	DiagnosticEnableCondition	1	ref	Reference to the EnableCondition which is mapped to a SWC service port with DiagnosticEnableConditionNeeds.
swcFlatServiceDependency	SwcServiceDependency	0..1	ref	Reference to a SwcServiceDependencyType that links ServiceNeeds to SWC service ports.
swcServiceDependency	SwcServiceDependency	0..1	iref	Instance reference to a SwcServiceDependency that links ServiceNeeds to SWC service ports.

Table 6.40: DiagnosticEnableConditionPortMapping

6.8.9 DiagnosticStorageCondition to Port Mapping

[TPS_DEXT_03019] Semantics of `DiagnosticStorageConditionPortMapping` ⌈ A `DiagnosticStorageConditionPortMapping` defines which SWC service port(s) have to be connected to which `DiagnosticStorageCondition`. This is realized by defining a `DiagnosticStorageConditionPortMapping` referencing a `DiagnosticStorageCondition` and an instance of `SwcServiceDependency`. ⌋*(RS_DEXT_00027, RS_DEXT_00052)*

If such an instance is not yet available, an ordinary reference to `SwcServiceDependency` can be given alternatively (i.e. without specifying a certain instance).

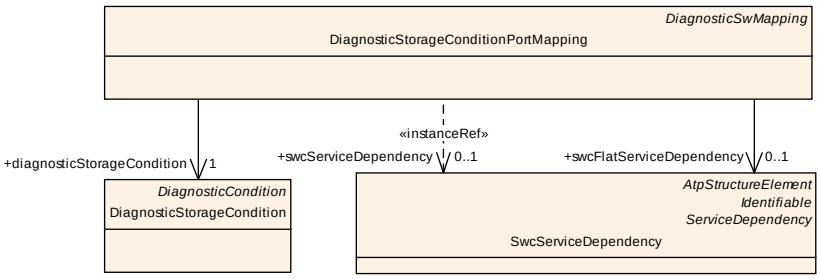

Figure 6.20: DiagnosticStorageConditionPortMapping

Class	**DiagnosticStorageConditionPortMapping**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticMapping			
Note	Defines to which SWC service ports with DiagnosticStorageConditionNeeds the DiagnosticStorageCondition is mapped. DiagnosticStorageConditionNeeds			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticMapping,DiagnosticSwMapping,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
diagnosticStorageCondition	DiagnosticStorageCondition	1	ref	Reference to the StorageCondition which is mapped to a SWC service port with DiagnosticStorageConditionNeeds.
swcFlatServiceDependency	SwcServiceDependency	0..1	ref	Reference to a SwcServiceDependencyType that links ServiceNeeds to SWC service ports.
swcServiceDependency	SwcServiceDependency	0..1	iref	Instance reference to a SwcServiceDependency that links ServiceNeeds to SWC service ports.

Table 6.41: DiagnosticStorageConditionPortMapping

6.8.10 Provided Data Mapping

[TPS_DEXT_03020] Semantics of `DiagnosticDemProvidedDataMapping` ⌈ The meta-class `DiagnosticDemProvidedDataMapping` does not seem to fulfill the condition for representing a mapping class because it only has one reference to a `DiagnosticDataElement` in the role `dataElement`.

However, the specific nature of this mapping is that the second element (the `DiagnosticDemProvidedDataMapping.dataProvider`) that is supposed to take place in the mapping cannot precisely be modeled as a single meta-class.

Therefore, there is no better way than to model the `DiagnosticDemProvidedDataMapping.dataProvider` by a `NameToken`. Of course, the collection of possible values of this attribute need to be agreed upon up-front, potentially on a project-specific basis.

The semantics of this mapping is to further qualify the access to the DiagnosticDataElement referenced in the role dataElement from within the Dem. ⌋(*RS_DEXT_00043*, *RS_DEXT_00052*)

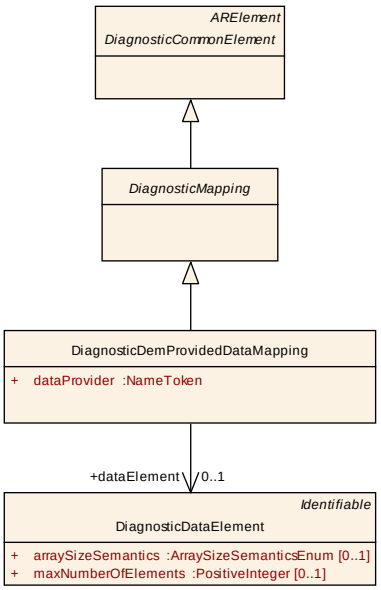

Figure 6.21: Modeling of the DiagnosticDemProvidedDataMapping

Class	**DiagnosticDemProvidedDataMapping**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::ServiceMapping			
Note	This represents the ability to define the nature of a data access for a DiagnsoticDataElement in the Dem.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,DiagnosticMapping,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
dataElement	DiagnosticDataElement	0..1	ref	This represents the DiagnosticDataElement for which the access is further qualified by the DiagnosticDemProvidedDataMapping.
dataProvider	NameToken	1	attr	This represents the ability to further specify the access within the Dem.

Table 6.42: DiagnosticDemProvidedDataMapping

6.9 DiagnosticOperationCycle

[TPS_DEXT_01087] Semantics of DiagnosticOperationCycle ⌈ Different types of DiagnosticOperationCycles are supported and defined by the type attribute, e.g. time between powering up and powering down the ECU or between ignition on and ignition off. ⌋(*RS_DEXT_00054*)

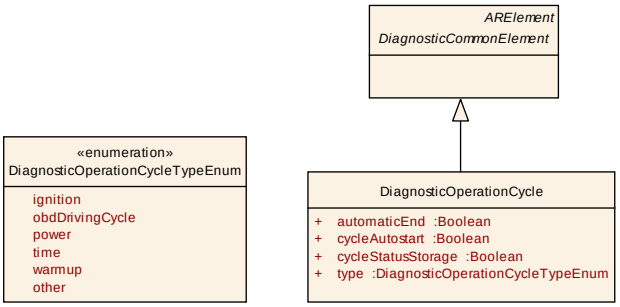

Figure 6.22: Modelling of DiagnosticOperationCycle

Class	**DiagnosticOperationCycle**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticOperationCycle			
Note	Definition of an operation cycle that is the base of the event qualifying and for Dem scheduling.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
automaticEnd	Boolean	1	attr	If set to true the driving cycle shall automatically end at either Dem_Shutdown() or Dem_Init().
cycleAutostart	Boolean	1	attr	This attribute defines if the operation cycles is automatically re-started during Dem_PreInit.
cycleStatusStorage	Boolean	1	attr	Defines if the operation cycle state is available over the power cycle (stored non-volatile) or not. true: the operation cycle state is stored non-volatile false: the operation cycle state is only stored volatile
type	DiagnosticOperationCycleTypeEnum	1	attr	Operation cycles types for the Dem to be supported by cycle-state APIs.

Table 6.43: DiagnosticOperationCycle

Enumeration	**DiagnosticOperationCycleTypeEnum**
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticOperationCycle
Note	Type of an operation cycle.
Literal	***Description***
ignition	Ignition ON / OFF cycle
obdDrivingCycle	OBD Driving cycle
other	further operation cycle
power	Power ON / OFF cycle
time	Time based operation cycle
warmup	OBD Warm up cycle

Table 6.44: DiagnosticOperationCycleTypeEnum

6.10 DiagnosticAging

[TPS_DEXT_03021] Aging ⌈ It is possible to remove a specific event from the event memory, if its fault conditions are not fulfilled for a certain period of time. This process is called as aging or unlearning.

This semantics is formalized by means of the meta-class `DiagnosticAging`. ⌋*(RS_DEXT_00055)*

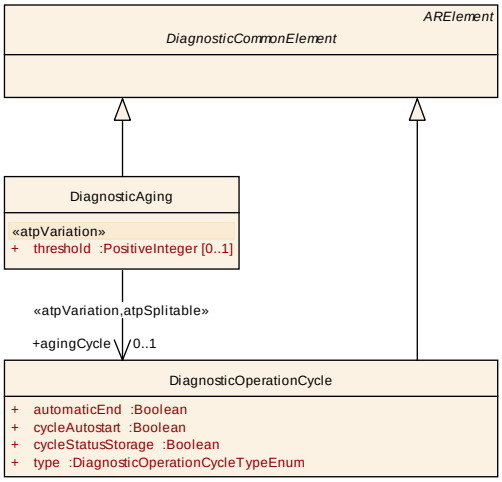

Figure 6.23: Modeling of `DiagnosticAging`

Class	**DiagnosticAging**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticAging			
Note	Defines the aging algorithm.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
agingCycle	DiagnosticOperationCycle	0..1	ref	This represents the applicable aging cycle. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=agingCycle, variationPoint.ShortLabel vh.latestBindingTime=preCompileTime
threshold	PositiveInteger	0..1	attr	Number of aging cycles needed to unlearn/delete the event. **Stereotypes:** atpVariation **Tags:** vh.latestBindingTime=preCompileTime

Table 6.45: DiagnosticAging

6.11 DiagnosticIndicator

[TPS_DEXT_03022] Different kinds of `DiagnosticIndicator`s ⌈ Different `type`s of Indicators can be defined with the `DiagnosticIndicator` element. For this, the attribute `DiagnosticIndicator.type` shall be used. ⌋*(RS_DEXT_00056)*

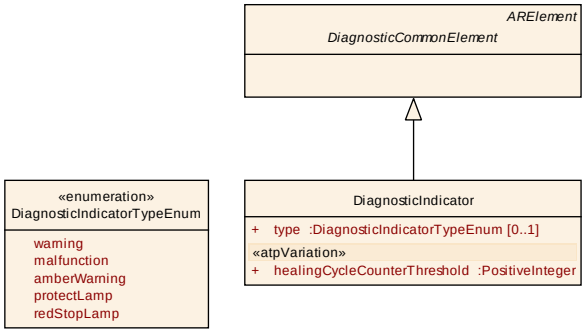

Figure 6.24: Modeling of `DiagnosticIndicator`

Class	**DiagnosticIndicator**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticIndicator			
Note	Definition of an indicator.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
healingCycleCounterThreshold	PositiveInteger	1	attr	This attribute defines the number of healing cycles for the WarningIndicatorOffCriteria **Stereotypes:** atpVariation **Tags:** vh.latestBindingTime=preCompileTime
type	DiagnosticIndicatorTypeEnum	0..1	attr	Defines the type of the indicator.

Table 6.46: DiagnosticIndicator

Enumeration	**DiagnosticIndicatorTypeEnum**
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticIndicator
Note	Type of an indicator.
Literal	***Description***
amberWarning	Amber Warning Lamp
malfunction	Malfunction Indicator Lamp
protectLamp	Protect Lamp
redStopLamp	Red Stop Lamp
warning	Warning

Table 6.47: DiagnosticIndicatorTypeEnum

A Mentioned Class Tables

For the sake of completeness, this chapter contains a set of class tables representing meta-classes mentioned in the context of this document but which are not contained directly in the scope of describing specific meta-model semantics.

Class	**ARElement (abstract)**			
Package	M2::AUTOSARTemplates::GenericStructure::GeneralTemplateClasses::ARPackage			
Note	An element that can be defined stand-alone, i.e. without being part of another element (except for packages of course).			
Base	ARObject,CollectableElement,Identifiable,MultilanguageReferrable,Packageable Element,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
–	–	–	–	–

Table A.1: ARElement

Class	**ARPackage**			
Package	M2::AUTOSARTemplates::GenericStructure::GeneralTemplateClasses::ARPackage			
Note	AUTOSAR package, allowing to create top level packages to structure the contained ARElements. ARPackages are open sets. This means that in a file based description system multiple files can be used to partially describe the contents of a package. This is an extended version of MSR's SW-SYSTEM.			
Base	ARObject,AtpBlueprint,AtpBlueprintable,Collectable Element,Identifiable,MultilanguageReferrable,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
arPackage	ARPackage	*	aggr	This represents a sub package within an ARPackage, thus allowing for an unlimited package hierarchy. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=shortName, variation Point.shortLabel vh.latestBindingTime=blueprintDerivationTime xml.sequenceOffset=30
element	PackageableEle ment	*	aggr	Elements that are part of this package **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=shortName, variation Point.shortLabel vh.latestBindingTime=systemDesignTime xml.sequenceOffset=20

Attribute	Datatype	Mul.	Kind	Note
referenceBase	ReferenceBase	*	aggr	This denotes the reference bases for the package. This is the basis for all relative references within the package. The base needs to be selected according to the base attribute within the references. **Stereotypes:** atpSplitable **Tags:** atp.Splitkey=shortLabel xml.sequenceOffset=10

Table A.2: ARPackage

Class	ApplicationPrimitiveDataType			
Package	M2::AUTOSARTemplates::SWComponentTemplate::Datatype::Datatypes			
Note	A primitive data type defines a set of allowed values. **Tags:** atp.recommendedPackage=ApplicationDataTypes			
Base	ARElement,ARObject,ApplicationDataType,AtpBlueprint,AtpBlueprintable,AtpClassifier,AtpType,AutosarDataType,CollectableElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	Datatype	Mul.	Kind	Note
–	–	–	–	–

Table A.3: ApplicationPrimitiveDataType

Class	ApplicationSwComponentType			
Package	M2::AUTOSARTemplates::SWComponentTemplate::Components			
Note	The ApplicationSwComponentType is used to represent the application software. **Tags:** atp.recommendedPackage=SwComponentTypes			
Base	ARElement,ARObject,AtomicSwComponentType,AtpBlueprint,AtpBlueprintable,AtpClassifier,AtpType,CollectableElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable,SwComponentType			
Attribute	Datatype	Mul.	Kind	Note
–	–	–	–	–

Table A.4: ApplicationSwComponentType

Class	AtomicSwComponentType (abstract)			
Package	M2::AUTOSARTemplates::SWComponentTemplate::Components			
Note	An atomic software component is atomic in the sense that it cannot be further decomposed and distributed across multiple ECUs.			
Base	ARElement,ARObject,AtpBlueprint,AtpBlueprintable,AtpClassifier,AtpType,CollectableElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable,SwComponentType			
Attribute	Datatype	Mul.	Kind	Note

Attribute	*Datatype*	*Mul.*	*Kind*	*Note*
internalBehavior	SwcInternalBehavior	0..1	aggr	The SwcInternalBehaviors owned by an AtomicSwComponentType can be located in a different physical file. Therefore the aggregation is «atpSplitable». **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=internalBehavior, variationPoint.shortLabel vh.latestBindingTime=preCompileTime
symbolProps	SymbolProps	0..1	aggr	This represents the SymbolProps for the AtomicSwComponentType. **Stereotypes:** atpSplitable **Tags:** atp.Splitkey=shortName

Table A.5: AtomicSwComponentType

Class	**AtpInstanceRef (abstract)**			
Package	M2::AUTOSARTemplates::GenericStructure::AbstractStructure			
Note	An M0 instance of a classifier may be represented as a tree rooted at that instance, where under each node come the sub-trees representing the instances which act as features under that node. An instance ref specifies a navigation path from any M0 tree-instance of the base (which is a classifier) to a leaf (which is an instance of the target).			
Base	ARObject			
Attribute	*Datatype*	*Mul.*	*Kind*	*Note*
atpBase	AtpClassifier	1	ref	This is the base from which the navigaion path starts. **Stereotypes:** atpAbstract; atpDerived
atpContextElement (ordered)	AtpPrototype	*	ref	This is one particular step in the navigation path. **Stereotypes:** atpAbstract
atpTarget	AtpFeature	1	ref	This is the target of the instance ref. In other words it is the terminal of the navigation path. **Stereotypes:** atpAbstract

Table A.6: AtpInstanceRef

Class	**BaseTypeDirectDefinition**			
Package	M2::AUTOSARTemplates::CommonStructure::BaseTypes			
Note	This BaseType is defined directly (as opposite to a derived BaseType)			
Base	ARObject,BaseTypeDefinition			
Attribute	*Datatype*	*Mul.*	*Kind*	*Note*

Attribute	*Datatype*	*Mul.*	*Kind*	*Note*
baseType Encoding	BaseTypeEncodingString	1	attr	This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **Tags:** xml.sequenceOffset=90
baseType Size	PositiveInteger	0..1	attr	Describes the length of the data type specified in the container in bits. **Tags:** xml.sequenceOffset=70
byteOrder	ByteOrderEnum	0..1	attr	This attribute specifies the byte order of the base type. **Tags:** xml.sequenceOffset=110
maxBaseTypeSize	PositiveInteger	0..1	attr	Describes the maximum length of the BaseType in bits. **Tags:** xml.sequenceOffset=80
memAlignment	PositiveInteger	0..1	attr	This attribute describes the alignment of the memory object in bits. E.g. "8" specifies, that the object in question is aligned to a byte while "32" specifies that it is aligned four byte. If the value is set to "0" the meaning shall be interpreted as "unspecified". **Tags:** xml.sequenceOffset=100

Attribute	Datatype	Mul.	Kind	Note

nativeDeclaration	NativeDeclarationString	0..1	attr	This attribute describes the declaration of such a base type in the native programming language, primarily in the Programming language C. This can then be used by a code generator to include the necessary declarations into a header file. For example
				
				BaseType with
				`shortName: "MyUnsignedInt"`
				`nativeDeclaration: "unsigned short"`
				
				Results in
				`typedef unsigned short MyUnsignedInt;`
				
				If the attribute is not defined the referring ImplementationDataTypes will not be generated as a typedef by RTE.
				
				If a nativeDeclaration type is given it shall fulfill the characteristic given by basetypeEncoding and baseTypeSize.
				
				This is required to ensure the consistent handling and interpretation by software components, RTE, COM and MCM systems.
				
				Tags: xml.sequenceOffset=120

Table A.7: BaseTypeDirectDefinition

Class	BswModuleDescription			
Package	M2::AUTOSARTemplates::BswModuleTemplate::BswOverview			
Note	Root element for the description of a single BSW module or BSW cluster. In case it describes a BSW module, the short name of this element equals the name of the BSW module. **Tags:** atp.recommendedPackage=BswModuleDescriptions			
Base	ARElement,ARObject,AtpBlueprint,AtpBlueprintable,AtpClassifier,AtpFeature,AtpStructureElement,CollectableElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
bswModuleDependency	BswModuleDependency	*	aggr	Describes the dependency to another BSW module. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=shortName, variationPoint.shortLabel vh.latestBindingTime=preCompileTime xml.sequenceOffset=20

Attribute	*Datatype*	*Mul.*	*Kind*	*Note*
bswModuleDocumentation	SwComponentDocumentation	0..1	aggr	This adds a documentation to the BSW module. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=bswModuleDocumentation, variationPoint.shortLabel vh.latestBindingTime=preCompileTime xml.sequenceOffset=6
internalBehavior	BswInternalBehavior	*	aggr	The various BswInternalBehaviors associated with a BswModuleDescription can be distributed over several physical files. Therefore the aggregation is «atpSplitable». **Stereotypes:** atpSplitable **Tags:** atp.Splitkey=shortName xml.sequenceOffset=65
moduleId	PositiveInteger	0..1	attr	Refers to the BSW Module Identifier defined by the AUTOSAR standard. For non-standardized modules, a proprietary identifier can be optionally chosen. **Tags:** xml.sequenceOffset=5
outgoingCallback	BswModuleEntry	*	ref	Specifies a callback, which will be called from this module if required by another module. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=outgoingCallback, variationPoint.shortLabel vh.latestBindingTime=preCompileTime xml.sequenceOffset=15
providedClientServerEntry	BswModuleClientServerEntry	*	aggr	Specifies that this module provides a client server entry which can be called from another parition or core.This entry is declared locally to this context and will be connected to the requiredClientServerEntry of another or the same module via the configuration of the BSW Scheduler. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=shortName, variationPoint.shortLabel vh.latestBindingTime=preCompileTime xml.sequenceOffset=45

Attribute	Datatype	Mul.	Kind	Note
providedData	VariableDataPrototype	*	aggr	Specifies a data prototype provided by this module in order to be read from another partition or core.The providedData is declared locally to this context and will be connected to the requiredData of another or the same module via the configuration of the BSW Scheduler. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=shortName, variationPoint.shortLabel vh.latestBindingTime=preCompileTime xml.sequenceOffset=55
providedEntry	BswModuleEntry	*	ref	Specifies an entry provided by this module which can be called by other modules. This includes "main" functions and interrupt routines, but not callbacks (because the signature of a callback is defined by the caller). **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=providedEntry, variationPoint.shortLabel vh.latestBindingTime=preCompileTime xml.sequenceOffset=10
providedModeGroup	ModeDeclarationGroupPrototype	*	aggr	A set of modes which is owned and provided by this module or cluster. It can be connected to the requiredModeGroups of other modules or clusters via the configuration of the BswScheduler. It can also be synchronized with modes provided via ports by an associated ServiceSwComponentType, EcuAbstractionSwComponentType or ComplexDeviceDriverSwComponentType. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=shortName, variationPoint.shortLabel vh.latestBindingTime=preCompileTime xml.sequenceOffset=25
releasedTrigger	Trigger	*	aggr	A Trigger released by this module or cluster. It can be connected to the requiredTriggers of other modules or clusters via the configuration of the BswScheduler. It can also be synchronized with Triggers provided via ports by an associated ServiceSwComponentType, EcuAbstractionSwComponentType or ComplexDeviceDriverSwComponentType. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=shortName, variationPoint.shortLabel vh.latestBindingTime=preCompileTime xml.sequenceOffset=35

Attribute	*Datatype*	*Mul.*	*Kind*	*Note*
requiredClientServerEntry	BswModuleClientServerEntry	*	aggr	Specifies that this module requires a client server entry which can be implemented on another parition or core.This entry is declared locally to this context and will be connected to the providedClientServerEntry of another or the same module via the configuration of the BSW Scheduler. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=shortName, variationPoint.shortLabel vh.latestBindingTime=preCompileTime xml.sequenceOffset=50
requiredData	VariableDataPrototype	*	aggr	Specifies a data prototype required by this module in oder to be provided from another partition or core.The requiredData is declared locally to this context and will be connected to the providedData of another or the same module via the configuration of the BswScheduler. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=shortName, variationPoint.shortLabel vh.latestBindingTime=preCompileTime xml.sequenceOffset=60
requiredModeGroup	ModeDeclarationGroupPrototype	*	aggr	Specifies that this module or cluster depends on a certain mode group. The requiredModeGroup is local to this context and will be connected to the providedModeGroup of another module or cluster via the configuration of the BswScheduler. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=shortName, variationPoint.shortLabel vh.latestBindingTime=preCompileTime xml.sequenceOffset=30
requiredTrigger	Trigger	*	aggr	Specifies that this module or cluster reacts upon an external trigger.This requiredTrigger is declared locally to this context and will be connected to the providedTrigger of another module or cluster via the configuration of the BswScheduler. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=shortName, variationPoint.shortLabel vh.latestBindingTime=preCompileTime xml.sequenceOffset=40

Table A.8: BswModuleDescription

Class	**BswServiceDependency**			
Package	M2::AUTOSARTemplates::BswModuleTemplate::BswBehavior			
Note	Specialization of ServiceDependency in the context of an BswInternalBehavior. It allows to associate BswModuleEntries and data defined for a BSW module or cluster to a given ServiceNeeds element.			
Base	ARObject,ServiceDependency			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
assignedData	RoleBasedDataAssignment	*	aggr	Defines the role of an associated data object (owned by this module or cluster) in the context of the ServiceNeeds element. **Stereotypes:** atpVariation **Tags:** vh.latestBindingTime=preCompileTime
assignedEntryRole	RoleBasedBswModuleEntryAssignment	*	aggr	Defines the role of an associated BswModuleEntry in the context of the ServiceNeeds element. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=assignedEntryRole, variationPoint.shortLabel vh.latestBindingTime=preCompileTime
ident	BswServiceDependencyIdent	0..1	aggr	This adds the ability to become referrable to BswServiceDependency. **Tags:** atp.Status=shallBecomeMandatory xml.sequenceOffset=-100
serviceNeeds	ServiceNeeds	1	aggr	The associated ServiceNeeds.

Table A.9: BswServiceDependency

Class	**ClientServerInterface**			
Package	M2::AUTOSARTemplates::SWComponentTemplate::PortInterface			
Note	A client/server interface declares a number of operations that can be invoked on a server by a client. **Tags:** atp.recommendedPackage=PortInterfaces			
Base	ARElement,ARObject,AtpBlueprint,AtpBlueprintable,AtpClassifier,AtpType,CollectableElement,Identifiable,MultilanguageReferrable,PackageableElement,PortInterface,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
operation	ClientServerOperation	1..*	aggr	ClientServerOperation(s) of this ClientServerInterface. **Stereotypes:** atpVariation **Tags:** vh.latestBindingTime=blueprintDerivationTime
possibleError	ApplicationError	*	aggr	Application errors that are defined as part of this interface.

Table A.10: ClientServerInterface

Class	≪atpVariation≫ **CommunicationCluster (abstract)**			
Package	M2::AUTOSARTemplates::SystemTemplate::Fibex::FibexCore::CoreTopology			
Note	The CommunicationCluster is the main element to describe the topological connection of communicating ECUs. A cluster describes the ensemble of ECUs, which are linked by a communication medium of arbitrary topology (bus, star, ring, ...). The nodes within the cluster share the same communication protocol, which may be event-triggered, time-triggered or a combination of both. A CommunicationCluster aggregates one or more physical channels. **Tags:** vh.latestBindingTime=postBuild			
Base	ARObject,CollectableElement,FibexElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
baudrate	PositiveUnlimitedInteger	0..1	attr	Channels speed in bits/s.
physicalChannel	PhysicalChannel	1..*	aggr	This relationship defines which channel element belongs to which cluster. A channel must be assigned to exactly one cluster, whereas a cluster may have one or more channels. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=shortName, variationPoint.shortLabel vh.latestBindingTime=systemDesignTime
protocolName	String	0..1	attr	The name of the protocol used.
protocolVersion	String	0..1	attr	The version of the protocol used.

Table A.11: CommunicationCluster

Class	**CompositionSwComponentType**			
Package	M2::AUTOSARTemplates::SWComponentTemplate::Composition			
Note	A CompositionSwComponentType aggregates SwComponentPrototypes (that in turn are typed by SwComponentTypes) as well as SwConnectors for primarily connecting SwComponentPrototypes among each others and towards the surface of the CompositionSwComponentType. By this means hierarchical structures of software-components can be created. **Tags:** atp.recommendedPackage=SwComponentTypes			
Base	ARElement,ARObject,AtpBlueprint,AtpBlueprintable,AtpClassifier,AtpType,CollectableElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable,SwComponentType			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***

Attribute	Datatype	Mul.	Kind	Note
component	SwComponentPrototype	*	aggr	The instantiated components that are part of this composition. The aggregation of SwComponentPrototype is subject to variability with the purpose to support the conditional existence of a SwComponentPrototype. Please be aware: if the conditional existence of SwComponentPrototypes is resolved post-build the deselected SwComponentPrototypes are still contained in the ECUs build but the instances are inactive in in that they are not scheduled by the RTE. The aggregation is marked as atpSplitable in order to allow the addition of service components to the ECU extract during the ECU integration. The use case for having 0 components owned by the CompositionSwComponentType could be to deliver an empty CompositionSwComponentType to e.g. a supplier for filling the internal structure. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=shortName, variationPoint.shortLabel vh.latestBindingTime=postBuild
connector	SwConnector	*	aggr	SwConnectors have the principal ability to establish a connection among PortPrototypes. They can have many roles in the context of a CompositionSwComponentType. Details are refined by subclasses. The aggregation of SwConnectors is subject to variability with the purpose to support variant data flow. The aggregation is marked as atpSplitable in order to allow the extension of the ECU extract with AssemblySwConnectors between ApplicationSwComponentTypes and ServiceSwComponentTypes during the ECU integration. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=shortName, variationPoint.shortLabel vh.latestBindingTime=postBuild
constantValueMapping	ConstantSpecificationMappingSet	*	ref	Reference to the ConstantSpecificationMapping to be applied for initValues of PPortComSpecs and RPortComSpec. **Stereotypes:** atpSplitable **Tags:** atp.Splitkey=constantValueMapping

Attribute	Datatype	Mul.	Kind	Note
dataTypeMapping	DataTypeMappingSet	*	ref	Reference to the DataTypeMapping to be applied for the used ApplicationDataTypes in PortInterfaces. Background: when developing subsystems it may happen that ApplicationDataTypes are used on the surface of CompositionSwComponentTypes. In this case it would be reasonable to be able to also provide the intended mapping to the ImplementationDataTypes. However, this mapping shall be informal and not technically binding for the implementers mainly because the RTE generator is not concerned about the CompositionSwComponentTypes. Rationale: if the mapping of ApplicationDataTypes on the delegated and inner PortPrototype matches then the mapping to ImplementationDataTypes is not impacting compatibility. **Stereotypes:** atpSplitable **Tags:** atp.Splitkey=dataTypeMapping
instantiationRTEEventProps	InstantiationRTEEventProps	*	aggr	This allows to define instantiation specific properties for RTE Events, in particular for instance specific scheduling. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=shortLabel, variationPoint.shortLabel vh.latestBindingTime=codeGenerationTime

Table A.12: CompositionSwComponentType

Class	CompuMethod			
Package	M2::AUTOSARTemplates::SWComponentTemplate::Datatype::ComputationMethod			
Note	This meta-class represents the ability to express the relationship between a physical value and the mathematical representation. Note that this is still independent of the technical implementation in data types. It only specifies the formula how the internal value corresponds to its physical pendant. **Tags:** atp.recommendedPackage=CompuMethods			
Base	ARElement,ARObject,AtpBlueprint,AtpBlueprintable,CollectableElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
compuInternalToPhys	Compu	0..1	aggr	This specifies the computation from internal values to physical values. **Tags:** xml.sequenceOffset=80

Attribute	Datatype	Mul.	Kind	Note
compuPhysToInternal	Compu	0..1	aggr	This represents the computation from physical values to the internal values. **Tags:** xml.sequenceOffset=90
displayFormat	DisplayFormatString	0..1	attr	This property specifies, how the physical value shall be displayed e.g. in documents or measurement and calibration tools. **Tags:** xml.sequenceOffset=20
unit	Unit	0..1	ref	This is the physical unit of the Physical values for which the CompuMethod applies. **Tags:** xml.sequenceOffset=30

Table A.13: CompuMethod

Class	DataConstr			
Package	M2::AUTOSARTemplates::CommonStructure::GlobalConstraints			
Note	This meta-class represents the ability to specify constraints on data. **Tags:** atp.recommendedPackage=DataConstrs			
Base	ARElement,ARObject,AtpBlueprint,AtpBlueprintable,CollectableElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
dataConstrRule	DataConstrRule	*	aggr	This is one particular rule within the data constraints. **Tags:** xml.roleElement=true; xml.roleWrapperElement=true; xml.sequenceOffset=30; xml.typeElement=false; xml.typeWrapperElement=false

Table A.14: DataConstr

Class	DiagEventDebounceAlgorithm (abstract)			
Package	M2::AUTOSARTemplates::CommonStructure::ServiceNeeds			
Note	This class represents the ability to specify the pre-debounce algorithm which is selected and/or required by the particular monitor. This class inherits from Identifiable in order to allow further documentation of the expected or implemented debouncing and to use the category for the identification of the expected / implemented debouncing.			
Base	ARObject,Identifiable,MultilanguageReferrable,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
–	–	–	–	–

Table A.15: DiagEventDebounceAlgorithm

Class	**DiagEventDebounceMonitorInternal**			
Package	M2::AUTOSARTemplates::CommonStructure::ServiceNeeds			
Note	This meta-class represents the ability to indicate that the pre-debounce algorithm shall be used by the DEM for this diagnostic monitor. This is related to setting the ECUC choice container DemDebounceAlgorithmClass to DemDebounceMonitorInternal. If the FaultDetectionAlogrithm is already known to be implemented by a specific BswModuleEntry the reference bswModuleEntry points to the function specification. If the FaultDetectionCounter value is accessible at a PortPrototype this PortPrototype shall be referenced by an assignedPort.			
Base	ARObject,DiagEventDebounceAlgorithm,Identifiable,MultilanguageReferrable,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
–	–	–	–	–

Table A.16: DiagEventDebounceMonitorInternal

Class	**DiagnosticEventNeeds**			
Package	M2::AUTOSARTemplates::CommonStructure::ServiceNeeds			
Note	Specifies the abstract needs on the configuration of the Diagnostic Event Manager for one diagnostic event. Its shortName can be regarded as a symbol identifying the diagnostic event from the viewpoint of the component or module which owns this element. In case the diagnostic event specifies a production error, the shortName shall be the name of the production error.			
Base	ARObject,DiagnosticCapabilityElement,Identifiable,MultilanguageReferrable,Referrable,ServiceNeeds			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
considerPtoStatus	Boolean	0..1	attr	PTO (Power Take Off) has an impact on the respective emission-related event (OBD). This information shall be provided by SW-C description in order to consider the PTO relevance e.g. for readiness (PID $01) computation. For events with dtcKind set to 'nonEmmissionRelatedDtc' this attribute is typically false.
deferringFid	FunctionInhibitionNeeds	*	ref	This reference contains the link to a function identifier within the FiM which is used by the monitor before delivering a result.
diagEventDebounceAlgorithm	DiagEventDebounceAlgorithm	0..1	aggr	Specifies the abstract need on the Debounce Algorithm applied by the Diagnostic Event Manager.
dtcKind	DtcKindEnum	0..1	attr	This attribute indicates the kind of the diagnostic monitor according to the SWS Diagnostic Event Manger. This attribute applies for the UDS diagnostics use case.

Attribute	Datatype	Mul.	Kind	Note
inhibitingFid	FunctionInhibitionNeeds	0..1	ref	This represents the primary Function Inhibition Identifier used for inhibition of the diagnostic monitor. The FID might either inhibit the monitoring of a symptom or the reporting of detected faults.
inhibitingSecondaryFid	FunctionInhibitionNeeds	*	ref	This represents the secondary Function Inhibition Identifier used for inhibition of the diagnostic monitor. Any of the FID inhibitions leads to an inhibition of the monitoring of a symptom or the reporting of detected faults.
obdDtcNumber	PositiveInteger	0..1	attr	This represents a reasonable Diagnostic Trouble Code. This allows to predefine the Diagnostic Trouble Code, e.g. if the a function developer has received a particular requirement from the OEM or from a standardization body. This attribute applies for the OBD diagnostics use case.
reportBehavior	ReportBehaviorEnum	0..1	attr	This switch indicates whether or not the BSW module is allowed to report the related Events before Dem_Init().
udsDtcNumber	PositiveInteger	0..1	attr	This represents a reasonable Diagnostic Trouble Code. This allows to predefine the Diagnostic Trouble Code, e.g. if the a function developer has received a particular requirement from the OEM or from a standardization body. This attribute applies for the UDS diagnostics use case.

Table A.17: DiagnosticEventNeeds

Class	DiagnosticIoControlNeeds			
Package	M2::AUTOSARTemplates::CommonStructure::ServiceNeeds			
Note	Specifies the general needs on the configuration of the Diagnostic Communication Manager (DCM) which are not related to a particular item (e.g. a PID). The main use case is the mapping of service ports to the DCM which are not related to a particular item.			
Base	ARObject,DiagnosticCapabilityElement,Identifiable,MultilanguageReferrable,Referrable,ServiceNeeds			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
currentValue	DiagnosticValueNeeds	0..1	ref	Reference to the DiagnosticValueNeeds indicating the access to the current value via signalBasedDiagnostics.
didNumber	PositiveInteger	0..1	attr	This represents a Data identifier for the diagnostic value. This allows to predefine the DID number if the a function developer has received a particular requirement from the OEM or from a standardization body.

Attribute	Datatype	Mul.	Kind	Note
freezeCurrentStateSupported	Boolean	0..1	attr	This attribute determines, if the referenced port supports temporary freezing of I/O value.
resetToDefaultSupported	Boolean	0..1	attr	This represents a flag for the existence of the ResetToDefault operation in the service interface.
shortTermAdjustmentSupported	Boolean	0..1	attr	This attribute determines, if the referenced port supports temporarily setting of I/O value to a specific value provided by the diagnostic tester.

Table A.18: DiagnosticIoControlNeeds

Class	DiagnosticParameter			
Package	M2::AUTOSARTemplates::DiagnosticExtract::CommonDiagnostics			
Note	This meta-class represents the ability to describe information relevant for the execution of a specific diagnostic service, i.e. it can be taken to parameterize the service.			
Base	ARObject			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
bitOffset	PositiveInteger	1	attr	This represents the bitOffset of the DiagnosticParameter
dataElement	DiagnosticDataElement	1	aggr	This represents the related dataElement of the DiagnosticParameter **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=shortName, variationPoint.shortLabel vh.latestBindingTime=postBuild

Table A.19: DiagnosticParameter

Class	DiagnosticRoutineNeeds			
Package	M2::AUTOSARTemplates::CommonStructure::ServiceNeeds			
Note	Specifies the general needs on the configuration of the Diagnostic Communication Manager (DCM) which are not related to a particular item (e.g. a PID). The main use case is the mapping of service ports to the DCM which are not related to a particular item.			
Base	ARObject,DiagnosticCapabilityElement,Identifiable,MultilanguageReferrable,Referrable,ServiceNeeds			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
diagRoutineType	DiagnosticRoutineTypeEnum	1	attr	This denotes the type of diagnostic routine which is implemented by the referenced server port.
ridNumber	PositiveInteger	0..1	attr	This represents a routine identifier for the diagnostic routine. This allows to predefine the RID number if the a function developer has received a particular requirement from the OEM or from a standardization body.

Table A.20: DiagnosticRoutineNeeds

Enumeration	**DiagnosticRoutineTypeEnum**
Package	M2::AUTOSARTemplates::CommonStructure::ServiceNeeds
Note	This enumerator specifies the different types of diagnostic routines.
Literal	***Description***
asynchronous	This indicates that the diagnostic server is not blocked while the diagnostic routine is running.
synchronous	This indicates that the diagnostic routine blocks the diagnostic server in the ECU while the routine is running.

Table A.21: DiagnosticRoutineTypeEnum

Class	**DiagnosticSecurityLevel**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dcm			
Note	This meta-class represents the ability to define a security level considered for diagnostic purposes.			
Base	ARElement,ARObject,CollectableElement,DiagnosticCommon Element,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
accessDataRecordSize	PositiveInteger	1	attr	This represents the size of the AccessDataRecord used in GetSeed. Unit:byte.
keySize	PositiveInteger	1	attr	This represents the size of the security key. Unit: byte.
numFailedSecurityAccess	PositiveInteger	1	attr	This represents the number of failed security accesses after which the delay time is activated.
securityDelayTime	TimeValue	1	attr	This represents the delay time after a failed security access. Unit: second.
seedSize	PositiveInteger	1	attr	This represents the size of the security seed. Unit: byte.

Table A.22: DiagnosticSecurityLevel

Class	**DiagnosticTroubleCodeJ1939**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::Dem::DiagnosticTroubleCode			
Note				
Base	ARElement,ARObject,CollectableElement,DiagnosticCommonElement,Diagnostic TroubleCode,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
j1939DtcValue	PositiveInteger	1	attr	Unique Diagnostic Trouble Code value for J1939 (consiting of SPN and FMI). **Stereotypes:** atpVariation **Tags:** vh.latestBindingTime=preCompileTime

Table A.23: DiagnosticTroubleCodeJ1939

Class	**DiagnosticValueNeeds**			
Package	M2::AUTOSARTemplates::CommonStructure::ServiceNeeds			
Note	Specifies the general needs on the configuration of the Diagnostic Communication Manager (DCM) which are not related to a particular item (e.g. a PID). The main use case is the mapping of service ports to the DCM which are not related to a particular item. In the case of using a sender receiver communicated value, the related value shall be taken via assignedData in the role "signalBasedDiagnostics". In case of using a client/server communicated value, the related value shall be communicated via the port referenced by asssignedPort. The details of this communication (e.g. appropriate naming conventions) are specified in the related software specifications (SWS).			
Base	ARObject,DiagnosticCapabilityElement,Identifiable,Multilanguage Referrable,Referrable,ServiceNeeds			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
dataLength	PositiveInteger	0..1	attr	This attribute is applicable only if the ServiceNeed is aggregated within BswModuleDependency. This attribute represents the length of data (in bytes) provided for this particular PID signal.
diagnostic ValueAcce ss	DiagnosticValue AccessEnum	0..1	attr	This attribute controls whether the data can be read and written or whether it is to be handled read-only.
didNumber	PositiveInteger	0..1	attr	This represents a Data identifier for the diagnostic value. This allows to predefine the DID number if the responsible function developer has received a particular requirement from the OEM or from a standardization body.
fixedLengt h	Boolean	0..1	attr	This attribute controls whether the data length of the data is fixed.
processing Style	DiagnosticProce ssingStyleEnum	0..1	attr	This attribute controls whether interaction requires the software-component to react synchronously on a request or whether it processes the request in background but still the DCM has to issue the call again to eventually obtain the result of the request.

Table A.24: DiagnosticValueNeeds

Class	≪**atpMixed**≫ **DocumentationBlock**			
Package	M2::AUTOSARTemplates::GenericStructure::GeneralTemplateClasses:: Documentation::BlockElements			
Note	This class represents a documentation block. It is made of basic text structure elements which can be displayed in a table cell.			
Base	ARObject			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***

Attribute	Datatype	Mul.	Kind	Note
defList	DefList	0..1	aggr	This represents a definition list in the documentation block. **Stereotypes:** atpVariation **Tags:** vh.latestBindingTime=postBuild xml.sequenceOffset=40
figure	MIFigure	0..1	aggr	This represents a figure in the documentation block. **Stereotypes:** atpVariation **Tags:** vh.latestBindingTime=postBuild xml.sequenceOffset=70
formula	MIFormula	0..1	aggr	This is a formula in the definition block. **Stereotypes:** atpVariation **Tags:** vh.latestBindingTime=postBuild xml.sequenceOffset=60
labeledList	LabeledList	0..1	aggr	This represents a labeled list. **Stereotypes:** atpVariation **Tags:** vh.latestBindingTime=postBuild xml.sequenceOffset=50
list	List	0..1	aggr	This represents numbered or unnumbered list. **Stereotypes:** atpVariation **Tags:** vh.latestBindingTime=postBuild xml.sequenceOffset=30
msrQuery P2	MsrQueryP2	0..1	aggr	
note	Note	0..1	aggr	This represents a note in the text flow. **Stereotypes:** atpVariation **Tags:** vh.latestBindingTime=postBuild xml.sequenceOffset=80
p	MultiLanguageP aragraph	0..1	aggr	This is one particular paragraph. **Stereotypes:** atpVariation **Tags:** vh.latestBindingTime=postBuild xml.sequenceOffset=10
structured Req	StructuredReq	0..1	aggr	This aggregation supports structured requirements embedded in a documentation block. **Stereotypes:** atpVariation **Tags:** vh.latestBindingTime=postBuild xml.sequenceOffset=100

Attribute	Datatype	Mul.	Kind	Note
trace	TraceableText	0..1	aggr	This represents traceable text in the documentation block. This allows to specify requirements/constraints in any documentation block. The kind of the trace is specified in the category. **Stereotypes:** atpVariation **Tags:** vh.latestBindingTime=postBuild xml.sequenceOffset=90
verbatim	MultiLanguageVerbatim	0..1	aggr	This represents one particular verbatim text. **Stereotypes:** atpVariation **Tags:** vh.latestBindingTime=postBuild xml.sequenceOffset=20

Table A.25: DocumentationBlock

Class	EcuInstance			
Package	M2::AUTOSARTemplates::SystemTemplate::Fibex::FibexCore::CoreTopology			
Note	ECUInstances are used to define the ECUs used in the topology. The type of the ECU is defined by a reference to an ECU specified with the ECU resource description. **Tags:** atp.recommendedPackage=EcuInstances			
Base	ARObject,CollectableElement,FibexElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
associatedComIPduGroup	ISignalIPduGroup	*	ref	With this reference it is possible to identify which ISignalIPduGroups are applicable for which CommunicationConnector/ ECU. Only top level ISignalIPduGroups shall be referenced by an EcuInstance. If an ISignalIPduGroup contains other ISignalIPduGroups than these contained ISignalIPduGroups shall not be referenced by the EcuInstance. Contained ISignalIPduGroups are associated to an EcuInstance via the top level ISignalIPduGroup.
associatedPdurIPduGroup	PdurIPduGroup	*	ref	With this reference it is possible to identify which PduR IPdu Groups are applicable for which CommunicationConnector/ ECU.
clientIdRange	ClientIdRange	0..1	aggr	Restriction of the Client Identifier for this Ecu to an allowed range of numerical values. The Client Identifier of the transaction handle is generated by the client RTE for inter-Ecu Client/Server communication.
comConfigurationGwTimeBase	TimeValue	0..1	attr	The period between successive calls to Com_MainFunctionRouteSignals of the AUTOSAR COM module in seconds.

Attribute	*Datatype*	*Mul.*	*Kind*	*Note*
comConfigurationRxTimeBase	TimeValue	0..1	attr	The period between successive calls to Com_MainFunctionRx of the AUTOSAR COM module in seconds.
comConfigurationTxTimeBase	TimeValue	0..1	attr	The period between successive calls to Com_MainFunctionTx of the AUTOSAR COM module in seconds.
comEnableMDTForCyclicTransmission	Boolean	0..1	attr	Enables for the Com module of this EcuInstance the minimum delay time monitoring for cyclic and repeated transmissions (TransmissionModeTiming has cyclicTiming assigned or eventControlledTiming with numberOfRepetitions > 0).
commController	CommunicationController	1..*	aggr	CommunicationControllers of the ECU.
connector	CommunicationConnector	*	aggr	All channels controlled by a single controller.
diagnosticAddress	Integer	0..1	attr	An ECU specific ID for responses of diagnostic routines.
diagnosticProps	DiagnosticEcuProps	0..1	aggr	This represents the diagnostic-related properties of an entire ECU.
partition	EcuPartition	*	aggr	Optional definition of Partitions within an Ecu.
pnResetTime	TimeValue	0..1	attr	Specifies the runtime of the reset timer in seconds. This reset time is valid for the reset of PN requests in the EIRA and in the ERA.
pncPrepareSleepTimer	TimeValue	0..1	attr	Time in seconds the PNC state machine shall wait in PNC_PREPARE_SLEEP.
sleepModeSupported	Boolean	1	attr	Specifies whether the ECU instance may be put to a "low power mode" • true: sleep mode is supported • false: sleep mode is not supported Note: This flag may only be set to "true" if the feature is supported by both hardware and basic software.
wakeUpOverBusSupported	Boolean	1	attr	Driver support for wakeup over Bus.

Table A.26: EcuInstance

Class	**ISignal**			
Package	M2::AUTOSARTemplates::SystemTemplate::Fibex::FibexCore::CoreCommunication			
Note	Signal of the Interaction Layer. The RTE supports a "signal fan-out" where the same System Signal is sent in different SignalIPdus to multiple receivers. To support the RTE "signal fan-out" each SignalIPdu contains ISignals. If the same System Signal is to be mapped into several SignalIPdus there is one ISignal needed for each ISignalToIPduMapping. ISignals describe the Interface between the Precompile configured RTE and the potentially Postbuild configured Com Stack (see ECUC Parameter Mapping). In case of the SystemSignalGroup an ISignal must be created for each SystemSignal contained in the SystemSignalGroup. **Tags:** atp.recommendedPackage=ISignals			
Base	ARObject,CollectableElement,FibexElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
dataTransformation	DataTransformation	0..1	ref	Optional reference to a DataTransformation which represents the transformer chain that is used to transform the data that shall be placed inside this ISignal. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=dataTransformation, variationPoint.shortLabel vh.latestBindingTime=codeGenerationTime
dataTypePolicy	DataTypePolicyEnum	1	attr	With the aggregation of SwDataDefProps an ISignal specifies how it is represented on the network. This representation follows a particular policy. Note that this causes some redundancy which is intended and can be used to support flexible development methodology as well as subsequent integrity checks. If the policy "networkRepresentationFromComSpec" is chosen the network representation from the ComSpec that is aggregated by the PortPrototype shall be used. If the "override" policy is chosen the requirements specified in the PortInterface and in the ComSpec are not fulfilled by the networkRepresentationProps. In case the System Description doesn't use a complete Software Component Description (VFB View) the "legacy" policy can be chosen.
iSignalProps	ISignalProps	0..1	aggr	Additional optional ISignal properties that may be stored in different files. **Stereotypes:** atpSplitable **Tags:** atp.Splitkey=iSignalProps

Attribute	Datatype	Mul.	Kind	Note
initValue	ValueSpecification	0..1	aggr	Optional definition of a ISignal's initValue in case the System Description doesn't use a complete Software Component Description (VFB View). This supports the inclusion of legacy system signals. This value can be used to configure the Signal's "InitValue". If a full DataMapping exist for the SystemSignal this information may be available from a configured SenderComSpec and ReceiverComSpec. In this case the initvalues in SenderComSpec and/or ReceiverComSpec override this optional value specification. Further restrictions apply from the RTE specification.
length	Integer	1	attr	Size of the signal in bits. The size needs to be derived from the mapped VariableDataPrototype according to the mapping of primitive DataTypes to BaseTypes as used in the RTE. Indicates maximum size for dynamic length signals. The ISignal length of zero bits is allowed.
networkRepresentationProps	SwDataDefProps	0..1	aggr	Specification of the actual network representation. The usage of SwDataDefProps for this purpose is restricted to the attributes compuMethod and baseType. The optional baseType attributes "memAllignment" and "byteOrder" shall not be used. The attribute "dataTypePolicy" in the SystemTemplate element defines whether this network representation shall be ignored and the information shall be taken over from the network representation of the ComSpec. If "override" is chosen by the system integrator the network representation can violate against the requirements defined in the PortInterface and in the network representation of the ComSpec. In case that the System Description doesn't use a complete Software Component Description (VFB View) this element is used to configure "ComSignalDataInvalidValue" and the Data Semantics.
systemSignal	SystemSignal	1	ref	Reference to the System Signal that is supposed to be transmitted in the ISignal.

Attribute	Datatype	Mul.	Kind	Note
transformationISignalProps	TransformationISignalProps	*	aggr	A transformer chain consists of an ordered list of transformers. The ISignal specific configuration properties for each transformer are defined in the TransformationISignalProps class. The transformer configuration properties that are common for all ISignals are described in the TransformationTechnology class.

Table A.27: ISignal

Class	ISignalIPduGroup			
Package	M2::AUTOSARTemplates::SystemTemplate::Fibex::FibexCore::CoreCommunication			
Note	The AUTOSAR COM Layer is able to start and to stop sending and receiving configurable groups of I-Pdus during runtime. An ISignalIPduGroup contains either ISignalIPdus or ISignalIPduGroups. **Tags:** atp.recommendedPackage=ISignaliPduGroup			
Base	ARObject,CollectableElement,FibexElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
communicationDirection	CommunicationDirectionType	1	attr	This attribute determines in which direction IPdus that are contained in this IPduGroup will be transmitted (communication direction can be either In or Out).
communicationMode	String	1	attr	This attribute defines the use-case for this ISignalIPduGroup (e.g. diagnostic, debugging etc.). For example, in a diagnostic mode all IPdus - which are not involved in diagnostic - are disabled. The use cases are not limited to a fixed enumeration and can be specified as a string.
containedISignalIPduGroup	ISignalIPduGroup	*	ref	An I-Pdu group can be included in other I-Pdu groups. Contained I-Pdu groups shall not be referenced by the EcuInstance.
iSignalIPdu	ISignalIPdu	*	ref	Reference to a set of Signal I-Pdus, which are contained in the ISignal I-Pdu Group. atpVariation: The content of a ISignal I-Pdu group can vary (->vehicle modes). **Stereotypes:** atpVariation **Tags:** vh.latestBindingTime=postBuild
nmPdu	NmPdu	*	ref	Reference to a set of NmPdus with NmUserData, which are contained in the ISignalIPduGroup. **Stereotypes:** atpVariation **Tags:** vh.latestBindingTime=postBuild

Table A.28: ISignalIPduGroup

Class	**IdentCaption (abstract)**			
Package	M2::AUTOSARTemplates::SWComponentTemplate::RPTScenario			
Note	This meta-class represents the caption. This allows having some meta classes optionally identifiable.			
Base	ARObject,AtpClassifier,AtpFeature,AtpStructureElement,Identifiable,MultilanguageReferrable,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
–	–	–	–	–

Table A.29: IdentCaption

Class	**Identifiable (abstract)**			
Package	M2::AUTOSARTemplates::GenericStructure::GeneralTemplateClasses::Identifiable			
Note	Instances of this class can be referred to by their identifier (within the namespace borders). In addition to this, Identifiables are objects which contribute significantly to the overall structure of an AUTOSAR description. In particular, Identifiables might contain Identifiables.			
Base	ARObject,MultilanguageReferrable,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
desc	MultiLanguageOverviewParagraph	0..1	aggr	This represents a general but brief (one paragraph) description what the object in question is about. It is only one paragraph! Desc is intended to be collected into overview tables. This property helps a human reader to identify the object in question. More elaborate documentation, (in particular how the object is built or used) should go to "introduction". **Tags:** xml.sequenceOffset=-60
category	CategoryString	0..1	attr	The category is a keyword that specializes the semantics of the Identifiable. It affects the expected existence of attributes and the applicability of constraints. **Tags:** xml.sequenceOffset=-50
adminData	AdminData	0..1	aggr	This represents the administrative data for the identifiable object. **Tags:** xml.sequenceOffset=-40
annotation	Annotation	*	aggr	Possibility to provide additional notes while defining a model element (e.g. the ECU Configuration Parameter Values). These are not intended as documentation but are mere design notes. **Tags:** xml.sequenceOffset=-25

Attribute	Datatype	Mul.	Kind	Note
introduction	Documentation Block	0..1	aggr	This represents more information about how the object in question is built or is used. Therefore it is a DocumentationBlock. **Tags:** xml.sequenceOffset=-30
uuid	String	0..1	attr	The purpose of this attribute is to provide a globally unique identifier for an instance of a meta-class. The values of this attribute should be globally unique strings prefixed by the type of identifier. For example, to include a DCE UUID as defined by The Open Group, the UUID would be preceded by "DCE:". The values of this attribute may be used to support merging of different AUTOSAR models. The form of the UUID (Universally Unique Identifier) is taken from a standard defined by the Open Group (was Open Software Foundation). This standard is widely used, including by Microsoft for COM (GUIDs) and by many companies for DCE, which is based on CORBA. The method for generating these 128-bit IDs is published in the standard and the effectiveness and uniqueness of the IDs is not in practice disputed. If the id namespace is omitted, DCE is assumed. An example is "DCE:2fac1234-31f8-11b4-a222-08002b34c003". The uuid attribute has no semantic meaning for an AUTOSAR model and there is no requirement for AUTOSAR tools to manage the timestamp. **Tags:** xml.attribute=true

Table A.30: Identifiable

Primitive	Identifier			
Package	M2::AUTOSARTemplates::GenericStructure::GeneralTemplateClasses::PrimitiveTypes			
Note	An Identifier is a string with a number of constraints on its appearance, satisfying the requirements typical programming languages define for their Identifiers. This datatype represents a string, that can be used as a c-Identifier. It shall start with a letter, may consist of letters, digits and underscores. **Tags:** xml.xsd.customType=IDENTIFIER; xml.xsd.maxLength=128; xml.xsd.pattern=[a-zA-Z][a-zA-Z0-9_]*; xml.xsd.type=string			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**

Attribute	*Datatype*	*Mul.*	*Kind*	*Note*
namePattern	String	0..1	attr	This attribute represents a pattern which shall be used to define the value of the identifier if the identifier in question is part of a blueprint. For more details refer to TPS_StandardizationTemplate. **Tags:** xml.attribute=true

Table A.31: Identifier

Class	**ImplementationDataType**			
Package	M2::AUTOSARTemplates::CommonStructure::ImplementationDataTypes			
Note	Describes a reusable data type on the implementation level. This will typically correspond to a typedef in C-code. **Tags:** atp.recommendedPackage=ImplementationDataTypes			
Base	ARElement,ARObject,AtpBlueprint,AtpBlueprintable,AtpClassifier,AtpType,AutosarDataType,CollectableElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
dynamicArraySizeProfile	String	0..1	attr	Specifies the profile which the array will follow in case this data type is a variable size array.
subElement (ordered)	ImplementationDataTypeElement	*	aggr	Specifies an element of an array, struct, or union data type. The aggregation of ImplementionDataTypeElement is subject to variability with the purpose to support the conditional existence of elements inside a ImplementationDataType representing a structure. **Stereotypes:** atpVariation **Tags:** vh.latestBindingTime=preCompileTime
symbolProps	SymbolProps	0..1	aggr	This represents the SymbolProps for the ImplementationDataType. **Stereotypes:** atpSplitable **Tags:** atp.Splitkey=shortName
typeEmitter	NameToken	0..1	attr	This attribute is used to control which part of the AUTOSAR toolchain is supposed to trigger data type definitions.

Table A.32: ImplementationDataType

Class	**InternalBehavior (abstract)**			
Package	M2::AUTOSARTemplates::CommonStructure::InternalBehavior			
Note	Common base class (abstract) for the internal behavior of both software components and basic software modules/clusters.			
Base	ARObject,AtpClassifier,AtpFeature,AtpStructureElement,Identifiable,MultilanguageReferrable,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
constantMemory	ParameterDataPrototype	*	aggr	Describes a read only memory object containing characteristic value(s) implemented by this InternalBehavior. The shortName of ParameterDataPrototype has to be equal to the ”C’ identifier of the described constant. The characteristic value(s) might be shared between SwComponentPrototypes of the same SwComponentType. The aggregation of constantMemory is subject to variability with the purpose to support variability in the software component or module implementations. Typically different algorithms in the implementation are requiring different number of memory objects. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=shortName, variationPoint.shortLabel vh.latestBindingTime=preCompileTime
constantValueMapping	ConstantSpecificationMappingSet	*	ref	Reference to the ConstanSpecificationMapping to be applied for the particular InternalBehavior **Stereotypes:** atpSplitable **Tags:** atp.Splitkey=constantValueMapping
dataTypeMapping	DataTypeMappingSet	*	ref	Reference to the DataTypeMapping to be applied for the particular InternalBehavior **Stereotypes:** atpSplitable **Tags:** atp.Splitkey=dataTypeMapping
exclusiveArea	ExclusiveArea	*	aggr	This specifies an ExclusiveArea for this InternalBehavior. The exclusiveArea is local to the component resp. module. The aggregation of ExclusiveAreas is subject to variability. Note: the number of ExclusiveAreas might vary due to the conditional existence of RunnableEntities or BswModuleEntities. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=shortName, variationPoint.shortLabel vh.latestBindingTime=preCompileTime

Attribute	Datatype	Mul.	Kind	Note
exclusiveAreaNestingOrder	ExclusiveAreaNestingOrder	*	aggr	This represents the set of ExclusiveAreaNestingOrder owned by the InternalBehavior. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=shortName, variationPoint.shortLabel vh.latestBindingTime=preCompileTime
staticMemory	VariableDataPrototype	*	aggr	Describes a read and writeable static memory object representing measurment variables implemented by this software component. Static is used in the meaning of non temporary and does not necessarily specify a linker encapsulation. This kind of memory is only supported if supportsMultipleInstantiation is FALSE. The shortName of the VariableDataPrototype has to be equal with the ”C’ identifier of the described variable. The aggregation of staticMemory is subject to variability with the purpose to support variability in the software components implementations. Typically different algorithms in the implementation are requiring different number of memory objects. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=shortName, variationPoint.shortLabel vh.latestBindingTime=preCompileTime

Table A.33: InternalBehavior

Class	MultiLanguageOverviewParagraph			
Package	M2::AUTOSARTemplates::GenericStructure::GeneralTemplateClasses::Documentation::TextModel::MultilanguageData			
Note	This is the content of a multilingual paragraph in an overview item.			
Base	ARObject			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
l2	LOverviewParagraph	1..*	aggr	This represents the text in one particular language. **Tags:** xml.roleElement=true; xml.roleWrapperElement=false; xml.sequenceOffset=20; xml.typeElement=false; xml.typeWrapperElement=false

Table A.34: MultiLanguageOverviewParagraph

Class	**MultilanguageReferrable (abstract)**			
Package	M2::AUTOSARTemplates::GenericStructure::GeneralTemplateClasses::Identifiable			
Note	Instances of this class can be referred to by their identifier (while adhering to namespace borders). They also may have a longName. But they are not considered to contribute substantially to the overall structure of an AUTOSAR description. In particular it does not contain other Referrables.			
Base	ARObject,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
longName	MultilanguageLongName	0..1	aggr	This specifies the long name of the object. Long name is targeted to human readers and acts like a headline.

Table A.35: MultilanguageReferrable

Primitive	**NameToken**
Package	M2::AUTOSARTemplates::GenericStructure::GeneralTemplateClasses::PrimitiveTypes
Note	This is an identifier as used in xml, e.g. xml-names. Basic difference to Identifier is the fact that it can contain "-". **Tags:** xml.xsd.customType=NMTOKEN-STRING; xml.xsd.type=NMTOKEN

Table A.36: NameToken

Class	**PPortPrototype**			
Package	M2::AUTOSARTemplates::SWComponentTemplate::Components			
Note	Component port providing a certain port interface.			
Base	ARObject,AbstractProvidedPortPrototype,AtpBlueprintable,AtpFeature,AtpPrototype,Identifiable,MultilanguageReferrable,PortPrototype,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
providedInterface	PortInterface	1	tref	The interface that this port provides. **Stereotypes:** isOfType

Table A.37: PPortPrototype

Class	**PortInterface (abstract)**			
Package	M2::AUTOSARTemplates::SWComponentTemplate::PortInterface			
Note	Abstract base class for an interface that is either provided or required by a port of a software component.			
Base	ARElement,ARObject,AtpBlueprint,AtpBlueprintable,AtpClassifier,AtpType,CollectableElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***

Attribute	Datatype	Mul.	Kind	Note
isService	Boolean	1	attr	This flag is set if the PortInterface is to be used for communication between an • ApplicationSwComponentType or • ServiceProxySwComponentType or • SensorActuatorSwComponentType or • ComplexDeviceDriverSwComponentType • ServiceSwComponentType • EcuAbstractionSwComponentType and a ServiceSwComponentType (namely an AUTOSAR Service) located on the same ECU. Otherwise the flag is not set.
serviceKind	ServiceProviderEnum	0..1	attr	This attribute provides further details about the nature of the applied service.

Table A.38: PortInterface

Class	PortPrototype (abstract)			
Package	M2::AUTOSARTemplates::SWComponentTemplate::Components			
Note	Base class for the ports of an AUTOSAR software component. The aggregation of PortPrototypes is subject to variability with the purpose to support the conditional existence of ports.			
Base	ARObject,AtpBlueprintable,AtpFeature,AtpPrototype,Identifiable,MultilanguageReferrable,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
clientServerAnnotation	ClientServerAnnotation	*	aggr	Annotation of this PortPrototype with respect to client/server communication.
delegatedPortAnnotation	DelegatedPortAnnotation	0..1	aggr	Annotations on this delegated port.
ioHwAbstractionServerAnnotation	IoHwAbstractionServerAnnotation	*	aggr	Annotations on this IO Hardware Abstraction port.
modePortAnnotation	ModePortAnnotation	*	aggr	Annotations on this mode port.
nvDataPortAnnotation	NvDataPortAnnotation	*	aggr	Annotations on this non voilatile data port.
parameterPortAnnotation	ParameterPortAnnotation	*	aggr	Annotations on this parameter port.
senderReceiverAnnotation	SenderReceiverAnnotation	*	aggr	Collection of annotations of this ports sender/receiver communication.

Attribute	Datatype	Mul.	Kind	Note
triggerPort Annotation	TriggerPortAnn otation	*	aggr	Annotations on this trigger port.

Table A.39: PortPrototype

Class	Referrable (abstract)			
Package	M2::AUTOSARTemplates::GenericStructure::GeneralTemplateClasses::Identifiable			
Note	Instances of this class can be referred to by their identifier (while adhering to namespace borders).			
Base	ARObject			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
shortName	Identifier	1	ref	This specifies an identifying shortName for the object. It needs to be unique within its context and is intended for humans but even more for technical reference. **Tags:** xml.enforceMinMultiplicity=true; xml.sequenceOffset=-100
shortName Fragment	ShortNameFrag ment	*	aggr	This specifies how the Referrable.shortName is composed of several shortNameFragments. **Tags:** xml.sequenceOffset=-90

Table A.40: Referrable

Class	RoleBasedPortAssignment			
Package	M2::AUTOSARTemplates::SWComponentTemplate::SwcInternalBehavior::Service Mapping			
Note	This class specifies an assignment of a role to a particular service port (RPortPrototype or PPortPrototype) of an AtomicSwComponentType. With this assignment, the role of the service port can be mapped to a specific ServiceNeeds element, so that a tool is able to create the correct connector.			
Base	ARObject			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
portPrototy pe	PortPrototype	1	ref	Service port used in the assigned role. This port shall either belong to the same AtomicSoftwareComponent as the SwcInternalBehavior which owns the ServiceDependency or to the same NvBlockComponentType as the NvBlockDescriptor.
role	Identifier	1	ref	This is the role of the assigned Port in the given context. The value shall be a shortName of the Blueprint of a PortInterface as standardized in the Software Specification of the related AUTOSAR Service.

Table A.41: RoleBasedPortAssignment

Class	**SenderReceiverInterface**			
Package	M2::AUTOSARTemplates::SWComponentTemplate::PortInterface			
Note	A sender/receiver interface declares a number of data elements to be sent and received. **Tags:** atp.recommendedPackage=PortInterfaces			
Base	ARElement,ARObject,AtpBlueprint,AtpBlueprintable,AtpClassifier,AtpType,CollectableElement,DataInterface,Identifiable,MultilanguageReferrable,PackageableElement,PortInterface,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
dataElement	VariableDataPrototype	1..*	aggr	The data elements of this SenderReceiverInterface.
invalidationPolicy	InvalidationPolicy	*	aggr	InvalidationPolicy for a particular dataElement

Table A.42: SenderReceiverInterface

Class	**ServiceNeeds (abstract)**			
Package	M2::AUTOSARTemplates::CommonStructure::ServiceNeeds			
Note	This expresses the abstract needs that a Software Component or Basic Software Module has on the configuration of an AUTOSAR Service to which it will be connected. "Abstract needs" means that the model abstracts from the Configuration Parameters of the underlying Basic Software.			
Base	ARObject,Identifiable,MultilanguageReferrable,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
–	–	–	–	–

Table A.43: ServiceNeeds

Class	**StructuredReq**			
Package	M2::AUTOSARTemplates::GenericStructure::GeneralTemplateClasses::Documentation::BlockElements::RequirementsTracing			
Note	This represents a structured requirement. This is intended for a case where specific requirements for features are collected. Note that this can be rendered as a labeled list.			
Base	ARObject,DocumentViewSelectable,Identifiable,MultilanguageReferrable,Paginateable,Referrable,Traceable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
conflicts	DocumentationBlock	0..1	aggr	This represents an informal specification of conflicts. **Tags:** xml.sequenceOffset=40
date	DateTime	1	attr	This represents the date when the requirement was initiated. **Tags:** xml.sequenceOffset=5

Attribute	*Datatype*	*Mul.*	*Kind*	*Note*
dependencies	Documentation Block	0..1	aggr	This represents an informal specifiaction of dependencies. Note that upstream tracing should be formalized in the property trace provided by the superclass Traceable. **Tags:** xml.sequenceOffset=30
description	Documentation Block	0..1	aggr	Ths represents the general description of the requirement. **Tags:** xml.sequenceOffset=10
importance	String	1	attr	This allows to represent the importance of the requirement. **Tags:** xml.sequenceOffset=8
issuedBy	String	1	attr	This represents the person, organization or authority which issued the requirement. **Tags:** xml.sequenceOffset=6
rationale	Documentation Block	0..1	aggr	This represents the rationale of the requirement. **Tags:** xml.sequenceOffset=20
remark	Documentation Block	0..1	aggr	This represents an informal remark. Note that this is not modeled as annotation, since these remark is still essential part of the requirement. **Tags:** xml.sequenceOffset=60
supporting Material	Documentation Block	0..1	aggr	This represents an informal specifiaction of the supporting material. **Tags:** xml.sequenceOffset=50
testedItem	Traceable	*	ref	This assocation represents the ability to trace on the same specification level. This supports for example the of acceptance tests. **Tags:** xml.sequenceOffset=70
type	String	1	attr	This attribute allows to denote the type of requirement to denote for example is it an "enhancement", "new feature" etc. **Tags:** xml.sequenceOffset=7
useCase	Documentation Block	0..1	aggr	This describes the relevant use cases. Note that formal references to use cases should be done in the trace relation. **Tags:** xml.sequenceOffset=35

Table A.44: StructuredReq

Class	**SwAxisGrouped**			
Package	M2::AUTOSARTemplates::CommonStructure::Axis			
Note	An SwAxisGrouped is an axis which is shared between multiple calibration parameters.			
Base	ARObject,SwCalprmAxisTypeProps			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
sharedAxisType	ApplicationPrimitiveDataType	0..1	ref	This is the datatype of the calibration parameter providing the shared axis.
swAxisIndex	AxisIndexType	0..1	attr	Describes which axis of the referenced calibration parameter provides the values for the group axis. The index satisfies the following convention: • 0 = value axis. in this case, the interpolation result of the referenced parameter is used as a base point index. This means that the A2L keyword CURVE_AXIS_REF can be supported. • The index should only be specified if the parameter under swCalprm contains more than one axis. It is standard practice for the axis index of parameters with more than one axis, to be set to 1, if data has not been assigned to swAxisIndex. **Tags:** xml.sequenceOffset=20
swCalprmRef	SwCalprmRefProxy	1	aggr	This property specifes the calibration parameter which serves as the input axis. In AUTOSAR, the type of the referenced Calibration parameter shall be compatible to the type specified by sharedAxisType. **Tags:** xml.roleElement=false; xml.roleWrapperElement=false; xml.sequenceOffset=30; xml.typeElement=false; xml.typeWrapperElement=false

Table A.45: SwAxisGrouped

Class	**SwAxisIndividual**			
Package	M2::AUTOSARTemplates::CommonStructure::Axis			
Note	This meta-class describes an axis integrated into a parameter (field etc.). The integration makes this individual to each parameter. The so-called grouped axis represents the counterpart to this. It is conceived as an independent parameter (see class SwAxisGrouped).			
Base	ARObject,SwCalprmAxisTypeProps			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
compuMethod	CompuMethod	0..1	ref	This is the compuMethod which is expected for the axis. It is used in early stages if the particular input-value is not yet available. **Tags:** xml.sequenceOffset=30

Attribute	*Datatype*	*Mul.*	*Kind*	*Note*
dataConstr	DataConstr	0..1	ref	Refers to constraints, e.g. for plausibility checks. **Tags:** xml.sequenceOffset=80
inputVariableType	ApplicationPrimitiveDataType	0..1	ref	This is the datatype of the input value for the axis. This allows to define e.g. a type of curve, where the input value is finalized at the access point.
swAxisGeneric	SwAxisGeneric	0..1	aggr	this specifies the properties of a generic axis if applicable. **Tags:** xml.sequenceOffset=90
swMaxAxisPoints	Integer	1	attr	Maximum number of base points contained in the axis of a map or curve. **Stereotypes:** atpVariation **Tags:** vh.latestBindingTime=preCompileTime xml.sequenceOffset=60
swMinAxisPoints	Integer	1	attr	Minimum number of base points contained in the axis of a map or curve. **Stereotypes:** atpVariation **Tags:** vh.latestBindingTime=preCompileTime xml.sequenceOffset=70

Attribute	Datatype	Mul.	Kind	Note
swVariableRef	SwVariableRefProxy	*	aggr	Refers to input variables of the axis. It is possible to specify more than one variable. Here the following is valid: • The variable with the highest priority shall be given first. It is used in the generation of the code and is also displayed first in the application system. • All variables referenced shall be of the same physical nature. This is usually detected in that the conversion formulae affected refer back to the same SI-units. In AUTOSAR this ensured by the constraint, that the referenced input variables shall use a type compatible to "inputVariableType". • This multiple referencing allows a base point distribution for more than one input variable to be used. One example of this are the temperature curves which can depend both on the induction air temperature and the engine temperature. These variables can be displayed simultaneously by MCD systems (adjustment systems), enabling operating points to be shown in the curves. **Tags:** xml.roleElement=false; xml.roleWrapperElement=true; xml.sequenceOffset=20; xml.typeElement=false; xml.typeWrapperElement=false
unit	Unit	0..1	ref	This represents the physical unit of the input value of the axis. It is provided to support the case that the particular input variable is not yet known. **Tags:** xml.sequenceOffset=40

Table A.46: SwAxisIndividual

Class	SwBaseType			
Package	M2::AUTOSARTemplates::CommonStructure::BaseTypes			
Note	This meta-class represents a base type used within ECU software. **Tags:** atp.recommendedPackage=BaseTypes			
Base	ARElement,ARObject,AtpBlueprint,AtpBlueprintable,BaseType,CollectableElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
–	–	–	–	–

Table A.47: SwBaseType

Class	**SwCalprmAxis**			
Package	M2::AUTOSARTemplates::CommonStructure::CalibrationParameter			
Note	This element specifies an individual input parameter axis (abscissa).			
Base	ARObject			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
category	CalprmAxisCategoryEnum	0..1	attr	This property specifies the category of a particular axis. **Tags:** xml.sequenceOffset=30
baseType	SwBaseType	0..1	ref	The SwBaseType to be used for the axis. Note that this is not applicable for ApplicationDataTypes. The value shall be ignored. **Tags:** atp.Status=removed xml.sequenceOffset=110
displayFormat	DisplayFormatString	0..1	attr	This property specifies how the axis values shall be displayed e.g. in documents or in measurement and calibration tools. **Tags:** xml.sequenceOffset=100
swAxisIndex	AxisIndexType	0..1	attr	This attribute specifies which axis is specified by the containing SwCalprmAxis. For example in a curve this is usually "1". In a map this is "1" or "2". **Tags:** xml.sequenceOffset=20
swCalibrationAccess	SwCalibrationAccessEnum	0..1	attr	Describes the applicability of parameters and variables. **Tags:** xml.sequenceOffset=90
swCalprmAxisTypeProps	SwCalprmAxisTypeProps	1	aggr	specific properties depending on the type of the axis. **Tags:** xml.roleElement=false; xml.roleWrapperElement=false; xml.sequenceOffset=40; xml.typeElement=true; xml.typeWrapperElement=false

Table A.48: SwCalprmAxis

Class	**SwCalprmAxisSet**			
Package	M2::AUTOSARTemplates::CommonStructure::CalibrationParameter			
Note	This element specifies the input parameter axes (abscissas) of parameters (and variables, if these are used adaptively).			
Base	ARObject			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
swCalprmAxis	SwCalprmAxis	*	aggr	One axis belonging to this SwCalprmAxisSet **Tags:** xml.roleElement=true; xml.roleWrapperElement=false; xml.sequenceOffset=20; xml.typeElement=false; xml.typeWrapperElement=false

Attribute	Datatype	Mul.	Kind	Note

Table A.49: SwCalprmAxisSet

Class	SwComponentType (abstract)			
Package	M2::AUTOSARTemplates::SWComponentTemplate::Components			
Note	Base class for AUTOSAR software components.			
Base	ARElement,ARObject,AtpBlueprint,AtpBlueprintable,AtpClassifier,AtpType,CollectableElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
consistencyNeeds	ConsistencyNeeds	*	aggr	This represents the colelction of ConsistencyNeeds owned by the enclosing SwComponentType. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=shortName, variationPoint.shortLabel vh.latestBindingTime=preCompileTime
port	PortPrototype	*	aggr	The ports through which this component can communicate. The aggregation of PortPrototype is subject to variability with the purpose to support the conditional existence of PortPrototypes. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=shortName, variationPoint.shortLabel vh.latestBindingTime=preCompileTime
portGroup	PortGroup	*	aggr	A port group being part of this component. **Stereotypes:** atpVariation **Tags:** vh.latestBindingTime=preCompileTime
swComponentDocumentation	SwComponentDocumentation	0..1	aggr	This adds a documentation to the SwComponentType. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=swComponentDocumentation, variationPoint.shortLabel vh.latestBindingTime=preCompileTime xml.sequenceOffset=-10
unitGroup	UnitGroup	*	ref	This allows for the specification of which UnitGroups are relevant in the context of referencing SwComponentType.

Table A.50: SwComponentType

Class	≪`atpVariation`≫ **SwDataDefProps**
Package	M2::AUTOSARTemplates::CommonStructure::DataDefProperties
Note	This class is a collection of properties relevant for data objects under various aspects. One could consider this class as a "pattern of inheritance by aggregation". The properties can be applied to all objects of all classes in which SwDataDefProps is aggregated. Note that not all of the attributes or associated elements are useful all of the time. Hence, the process definition (e.g. expressed with an OCL or a Document Control Instance MSR-DCI) has the task of implementing limitations. SwDataDefProps covers various aspects: • Structure of the data element for calibration use cases: is it a single value, a curve, or a map, but also the recordLayouts which specify how such elements are mapped/converted to the DataTypes in the programming language (or in AUTOSAR). This is mainly expressed by properties like swRecordLayout and swCalprmAxisSet • Implementation aspects, mainly expressed by swImplPolicy, swVariableAccessImplPolicy, swAddrMethod, swPointerTagetProps, baseType, implementationDataType and additionalNativeTypeQualifier • Access policy for the MCD system, mainly expressed by swCalibrationAccess • Semantics of the data element, mainly expressed by compuMethod and/or unit, dataConstr, invalidValue • Code generation policy provided by swRecordLayout **Tags:** vh.latestBindingTime=codeGenerationTime
Base	ARObject

Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
additionalNativeTypeQualifier	NativeDeclarationString	0..1	attr	This attribute is used to declare native qualifiers of the programming language which can neither be deduced from the baseType (e.g. because the data object describes a pointer) nor from other more abstract attributes. Examples are qualifiers like "volatile", "strict" or "enum" of the C-language. All such declarations have to be put into one string. **Tags:** xml.sequenceOffset=235
annotation	Annotation	*	aggr	This aggregation allows to add annotations (yellow pads ...) related to the current data object. **Tags:** xml.roleElement=true; xml.roleWrapperElement=true; xml.sequenceOffset=20; xml.typeElement=false; xml.typeWrapperElement=false
baseType	SwBaseType	0..1	ref	Base type associated with the containing data object. **Tags:** xml.sequenceOffset=50

Attribute	Datatype	Mul.	Kind	Note
compuMethod	CompuMethod	0..1	ref	Computation method associated with the semantics of this data object. **Tags:** xml.sequenceOffset=180
dataConstr	DataConstr	0..1	ref	Data constraint for this data object. **Tags:** xml.sequenceOffset=190
displayFormat	DisplayFormatString	0..1	attr	This property describes how a number is to be rendered e.g. in documents or in a measurement and calibration system. **Tags:** xml.sequenceOffset=210
implementationDataType	ImplementationDataType	0..1	ref	This association denotes the ImplementationDataType of a data declaration via its aggregated SwDataDefProps. It is used whenever a data declaration is not directly referring to a base type. Especially • redefinition of an ImplementationDataType via a "typedef" to another ImplementationDatatype • the target type of a pointer (see SwPointerTargetProps), if it does not refer to a base type directly • the data type of an array or record element within an ImplementationDataType, if it does not refer to a base type directly • the data type of an SwServiceArg, if it does not refer to a base type directly **Tags:** xml.sequenceOffset=215
invalidValue	ValueSpecification	0..1	aggr	Optional value to express invalidity of the actual data element. **Tags:** xml.sequenceOffset=255
stepSize	Float	0..1	attr	This attribute can be used to define a value which is added to or subtracted from the value of a DataPrototype when using up/down keys while calibrating.
swAddrMethod	SwAddrMethod	0..1	ref	Addressing method related to this data object. Via an association to the same SwAddrMethod it can be specified that several DataPrototypes shall be located in the same memory without already specifying the memory section itself. **Tags:** xml.sequenceOffset=30

Attribute	*Datatype*	*Mul.*	*Kind*	*Note*
swAlignment	AlignmentType	0..1	attr	The attribute describes the intended alignment of the DataPrototype. If the attribute is not defined the alignment is determined by the swBaseType size and the memoryAllocationKeywordPolicy of the referenced SwAddrMethod. **Tags:** xml.sequenceOffset=33
swBitRepresentation	SwBitRepresentation	0..1	aggr	Description of the binary representation in case of a bit variable. **Tags:** xml.sequenceOffset=60
swCalibrationAccess	SwCalibrationAccessEnum	0..1	attr	Specifies the read or write access by MCD tools for this data object. **Tags:** xml.sequenceOffset=70
swCalprmAxisSet	SwCalprmAxisSet	0..1	aggr	This specifies the properties of the axes in case of a curve or map etc. This is mainly applicable to calibration parameters. **Tags:** xml.sequenceOffset=90
swComparisonVariable	SwVariableRefProxy	*	aggr	Variables used for comparison in an MCD process. **Tags:** xml.sequenceOffset=170; xml.typeElement=false
swDataDependency	SwDataDependency	0..1	aggr	Describes how the value of the data object has to be calculated from the value of another data object (by the MCD system). **Tags:** xml.sequenceOffset=200
swHostVariable	SwVariableRefProxy	0..1	aggr	Contains a reference to a variable which serves as a host-variable for a bit variable. Only applicable to bit objects. **Tags:** xml.sequenceOffset=220; xml.typeElement=false
swImplPolicy	SwImplPolicyEnum	0..1	attr	Implementation policy for this data object. **Tags:** xml.sequenceOffset=230

Attribute	Datatype	Mul.	Kind	Note
swIntendedResolution	Numerical	0..1	attr	The purpose of this element is to describe the requested quantization of data objects early on in the design process. The resolution ultimately occurs via the conversion formula present (compuMethod), which specifies the transition from the physical world to the standardized world (and vice-versa) (here, "the slope per bit" is present implicitly in the conversion formula). In the case of a development phase without a fixed conversion formula, a pre-specification can occur through swIntendedResolution. The resolution is specified in the physical domain according to the property "unit". **Tags:** xml.sequenceOffset=240
swInterpolationMethod	Identifier	0..1	ref	This is a keyword identifying the mathematical method to be applied for interpolation. The keyword needs to be related to the interpolation routine which needs to be invoked. **Tags:** xml.sequenceOffset=250
swIsVirtual	Boolean	0..1	attr	This element distinguishes virtual objects. Virtual objects do not appear in the memory, their derivation is much more dependent on other objects and hence they shall have a swDataDependency . **Tags:** xml.sequenceOffset=260
swPointerTargetProps	SwPointerTargetProps	0..1	aggr	Specifies that the containing data object is a pointer to another data object. **Tags:** xml.sequenceOffset=280
swRecordLayout	SwRecordLayout	0..1	ref	Record layout for this data object. **Tags:** xml.sequenceOffset=290
swRefreshTiming	MultidimensionalTime	0..1	aggr	This element specifies the frequency in which the object involved shall be or is called or calculated. This timing can be collected from the task in which write access processes to the variable run. But this cannot be done by the MCD system. So this attribute can be used in an early phase to express the desired refresh timing and later on to specify the real refresh timing. **Tags:** xml.sequenceOffset=300

Attribute	Datatype	Mul.	Kind	Note
swTextProps	SwTextProps	0..1	aggr	the specific properties if the data object is a text object. **Tags:** xml.sequenceOffset=120
swValueBlockSize	Numerical	0..1	attr	This represents the size of a Value Block **Stereotypes:** atpVariation **Tags:** vh.latestBindingTime=preCompileTime xml.sequenceOffset=80
unit	Unit	0..1	ref	Physical unit associated with the semantics of this data object. This attribute applies if no compuMethod is specified. If both units (this as well as via compuMethod) are specified the units shall be compatible. **Tags:** xml.sequenceOffset=350
valueAxisDataType	ApplicationPrimitiveDataType	0..1	ref	The referenced ApplicationPrimitiveDataType represents the primitive data type of the value axis within a compound primitive (e.g. curve, map). It supersedes CompuMethod, Unit, and BaseType. **Tags:** xml.sequenceOffset=355

Table A.51: SwDataDefProps

Class	SwcServiceDependency			
Package	M2::AUTOSARTemplates::SWComponentTemplate::SwcInternalBehavior::ServiceMapping			
Note	Specialization of ServiceDependency in the context of an SwcInternalBehavior. It allows to associate ports, port groups and (in special cases) data defined for an atomic software component to a given ServiceNeeds element.			
Base	ARObject,AtpClassifier,AtpFeature,AtpStructureElement,Identifiable,MultilanguageReferrable,Referrable,ServiceDependency			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
assignedData	RoleBasedDataAssignment	*	aggr	Defines the role of an associated data object of the same component. **Stereotypes:** atpVariation **Tags:** vh.latestBindingTime=preCompileTime
assignedPort	RoleBasedPortAssignment	*	aggr	Defines the role of an associated port of the same component. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=assignedPort, variationPoint.shortLabel vh.latestBindingTime=preCompileTime

Attribute	Datatype	Mul.	Kind	Note
representedPortGroup	PortGroup	0..1	ref	This reference specifies an association between the ServiceNeeeds and a PortGroup, for example to request a communication mode which applies for communication via these ports. The referred PortGroup shall be local to this atomic SWC, but via the links between the PortGroups, a tool can evaluate this information such that all the ports linked via this port group on the same ECU can be found.
serviceNeeds	ServiceNeeds	1	aggr	The associated ServiceNeeds.

Table A.52: SwcServiceDependency

Class	System			
Package	M2::AUTOSARTemplates::SystemTemplate			
Note	The top level element of the System Description. The System description defines five major elements: Topology, Software, Communication, Mapping and Mapping Constraints. The System element directly aggregates the elements describing the Software, Mapping and Mapping Constraints; it contains a reference to an ASAM FIBEX description specifying Communication and Topology. **Tags:** atp.recommendedPackage=Systems			
Base	ARElement,ARObject,AtpClassifier,AtpFeature,AtpStructureElement,CollectableElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	**Datatype**	**Mul.**	**Kind**	**Note**
clientIdDefinitionSet	ClientIdDefinitionSet	*	ref	Set of Client Identifiers that are used for inter-ECU client-server communication in the System.
containerIPduHeaderByteOrder	ByteOrderEnum	0..1	attr	Defines the byteOrder of the header in ContainerIPdus.
ecuExtractVersion	RevisionLabelString	0..1	attr	Version number of the Ecu Extract.
fibexElement	FibexElement	*	ref	Reference to ASAM FIBEX elements specifying Communication and Topology. All Fibex Elements used within a System Description shall be referenced from the System Element. atpVariation: In order to describe a product-line, all FibexElements can be optional. **Stereotypes:** atpVariation **Tags:** vh.latestBindingTime=postBuild

Attribute	Datatype	Mul.	Kind	Note
mapping	SystemMapping	*	aggr	Aggregation of all mapping aspects (mapping of SW components to ECUs, mapping of data elements to signals, and mapping constraints). In order to support OEM / Tier 1 interaction and shared development for one common System this aggregation is atpSplitable and atpVariation. The content of SystemMapping can be provided by several parties using different names for the SystemMapping. This element is not required when the System description is used for a network-only use-case. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=shortName, variation Point.shortLabel vh.latestBindingTime=postBuild
pncVector Length	PositiveInteger	0..1	attr	Length of the partial networking request release information vector (in bytes).
pncVector Offset	PositiveInteger	0..1	attr	Absolute offset (with respect to the NM-PDU) of the partial networking request release information vector that is defined in bytes as an index starting with 0.
rootSoftwa reComposi tion	RootSwCompos itionPrototype	0..1	aggr	Aggregation of the root software composition, containing all software components in the System in a hierarchical structure. This element is not required when the System description is used for a network-only use-case. atpVariation: The RootSwCompositionPrototype can vary. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=shortName, variation Point.shortLabel vh.latestBindingTime=systemDesignTime
systemDoc umentation	Chapter	*	aggr	Possibility to provide additional documentation while defining the System. The System documentation can be composed of several chapters. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=shortName, variation Point.shortLabel vh.latestBindingTime=systemDesignTime xml.sequenceOffset=-10
systemVer sion	RevisionLabelSt ring	1	attr	Version number of the System Description.

Table A.53: System

Class	**TraceableText**			
Package	M2::AUTOSARTemplates::GenericStructure::GeneralTemplateClasses::Documentation::BlockElements::RequirementsTracing			
Note	This meta-class represents the ability to denote a traceable text item such as requirements etc. The following approach appliles: • **shortName** represents the tag for tracing • **longName** represents the head line • **category** represents the kind of the tagged text			
Base	ARObject,DocumentViewSelectable,Identifiable,MultilanguageReferrable,Paginateable,Referrable,Traceable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
text	Documentation Block	1	aggr	This represents the text to which the tag applies. **Tags:** xml.roleElement=false; xml.roleWrapperElement=false; xml.sequenceOffset=30; xml.typeElement=false; xml.typeWrapperElement=false

Table A.54: TraceableText

Class	**Unit**			
Package	M2::AUTOSARTemplates::SWComponentTemplate::Datatype::Units			
Note	This is a physical measurement unit. All units that might be defined should stem from SI units. In order to convert one unit into another factor and offset are defined. For the calculation from SI-unit to the defined unit the factor (factorSiToUnit) and the offset (offsetSiToUnit) are applied: unit = siUnit * factorSiToUnit + offsetSiToUnit For the calculation from a unit to SI-unit the reciprocal of the factor (factorSiToUnit) and the negation of the offset (offsetSiToUnit) are applied: siUnit = (unit - offsetSiToUnit) / factorSiToUnit **Tags:** atp.recommendedPackage=Units			
Base	ARElement,ARObject,CollectableElement,Identifiable,MultilanguageReferrable,PackageableElement,Referrable			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
displayName	SingleLanguageUnitNames	0..1	aggr	This specifies how the unit shall be displayed in documents or in user interfaces of tools.The displayName corresponds to the Unit.Display in an ASAM MCD-2MC file. **Tags:** xml.sequenceOffset=20
factorSiToUnit	Float	0..1	attr	This is the factor for the conversion from and to siUnits. **Tags:** xml.sequenceOffset=30

Attribute	*Datatype*	*Mul.*	*Kind*	*Note*
offsetSiToUnit	Float	0..1	attr	This is the offset for the conversion from and to siUnits. **Tags:** xml.sequenceOffset=40
physicalDimension	PhysicalDimension	0..1	ref	This association represents the physical dimension to which the unit belongs to. Note that only values with units of the same physical dimensions might be converted. **Tags:** xml.sequenceOffset=50

Table A.55: Unit

B History of Constraints and Specification Items

B.1 Constraint History of this Document according to AUTOSAR R4.2.1

B.1.1 Added Specification Items in R4.2.1

Please note that specification items listed using an bold typeface have been deleted in later versions of the document.

Number	Heading
[TPS_DEXT_01000]	AUTOSAR diagnostics supports two kinds of data identifiers
[TPS_DEXT_01001]	Definition of a fixed-sized array
[TPS_DEXT_01002]	Definition of a variable-sized array
[TPS_DEXT_01003]	`DiagnosticContributionSet` is the central part of the `DiagnosticExtract`
[TPS_DEXT_01004]	`DiagnosticContributionSet` defines the scope of the `DiagnosticExtract`
[TPS_DEXT_01005]	`DiagnosticContributionSet` can exist independently
[TPS_DEXT_01006]	The role of `DiagnosticServiceTable`s in the context of a `DiagnosticContributionSet`
[TPS_DEXT_01007]	Common properties of a `DiagnosticExtract`
[TPS_DEXT_01008]	`DiagnosticContributionSet` defines the scope for the application of the common diagnostic properties
[TPS_DEXT_01009]	Limited support for the configuration of custom diagnostic services
[TPS_DEXT_01010]	Configuration of custom diagnostic services
[TPS_DEXT_01011]	Semantics of `DiagnosticSession.id`
[TPS_DEXT_01012]	Rationale for the modeling of the multiplicity of `DiagnosticAccessPermission.securityLevel`
[TPS_DEXT_01013]	Specification of sub-functions by means of attribute `DiagnosticServiceInstance.category`
[TPS_DEXT_01014]	Possible values of the `category` attribute for diagnostic services
[TPS_DEXT_01015]	Meaning of attributes of `DiagnosticIOControl`
[TPS_DEXT_01016]	The capability returnControlToEcu
[TPS_DEXT_01017]	Meaning of `DiagnosticIOControl.dataIdentifier`
[TPS_DEXT_01018]	`InputOutput Control` does not define any sub-functions

[TPS_DEXT_01019]	Correspondence of category values to numerical values mentioned in the ISO 14229-1
[TPS_DEXT_01020]	Manufacturer-specific values for sub-functions of service EcuReset
[TPS_DEXT_01021]	Semantics of DiagnosticEcuReset.customSubFunctionNumber
[TPS_DEXT_01022]	ClearDiagnosticInformation does not define any sub-functions
[TPS_DEXT_01023]	WriteMemoryByAddress does not define any sub-functions
[TPS_DEXT_01024]	ReadMemoryByAddress does not define any sub-functions
[TPS_DEXT_01025]	TransferExit does not define any sub-functions
[TPS_DEXT_01026]	DataTransfer does not define any sub-functions
[TPS_DEXT_01027]	RequestDownload does not define any sub-functions
[TPS_DEXT_01028]	RequestUpload does not define any sub-functions
[TPS_DEXT_01029]	Correspondence of category values to numerical values mentioned in the ISO 14229-1
[TPS_DEXT_01030]	Manufacturer-specific values for sub-functions of service CommunicationControl
[TPS_DEXT_01031]	Semantics of DiagnosticComControl.customSubFunctionNumber
[TPS_DEXT_01032]	Impact of the DiagnosticComControlClass on the state management for CommunicationClusters
[TPS_DEXT_01033]	Semantics of triggers in the context of a DiagnosticResponseOnEvent
[TPS_DEXT_01034]	Sub-functions of the service ReadDTCInformation
[TPS_DEXT_01035]	Existence of DiagnosticRoutine.stop and DiagnosticRoutine.requestResult
[TPS_DEXT_01036]	Work-flow within the execution of the diagnostic service SecurityAccess
[TPS_DEXT_01037]	Semantics of DiagnosticSecurityAccess.requestSeedId
[TPS_DEXT_01038]	Motivation for making the reference DiagnosticSecurityAccess.securityLevel ≪atpSplitable≫
[TPS_DEXT_01039]	Identification of the sub-function of DiagnosticSessionControl
[TPS_DEXT_01040]	Use case where the DiagnosticExtract refers to software-components
[TPS_DEXT_01041]	Semantics of attribute DiagnosticServiceDataMapping.diagnosticDataElement
[TPS_DEXT_01042]	Dem uses DiagnosticServiceDataMapping
[TPS_DEXT_01043]	Purpose of DiagnosticServiceSwMapping
[TPS_DEXT_01044]	BswServiceDependency needs to act as the target of a reference
[TPS_DEXT_01045]	Supported diagnostic services
[TPS_DEXT_01046]	ECU configuration is not suitable to be exchanged between partners in an ECU development project
[TPS_DEXT_01047]	Differences in the development processes for diagnostics at automotive OEMs and ECU suppliers
[TPS_DEXT_01048]	Actual algorithm for the diagnostic event debouncing
[TPS_DEXT_01049]	Consistency of DiagnosticServiceSwMapping with respect to routine IDs
[TPS_DEXT_01050]	Consistency of DiagnosticServiceSwMapping with respect to data IDs
[TPS_DEXT_01051]	Consistency of DiagnosticServiceSwMapping with respect to data IDs
[TPS_DEXT_01052]	Existence of attribute DiagnosticServiceInstance.accessPermission
[TPS_DEXT_01053]	Existence of DiagnosticSecurityAccess.securityLevel
[TPS_DEXT_01054]	Existence of DiagnosticDataByIdentifier.dataIdentifier
[TPS_DEXT_01055]	Standardized values of DiagnosticContributionSet.category
[TPS_DEXT_01056]	Applicable values for DiagnosticEcuReset.category
[TPS_DEXT_01057]	Allowed values of DiagnosticComControl.category
[TPS_DEXT_01058]	Standardized values for DiagnosticDynamicallyDefineDataIdentifier.category
[TPS_DEXT_01059]	Applicable values for DiagnosticPeriodicRate.category
[TPS_DEXT_01060]	Applicable values for DiagnosticReadDTCInformation.category
[TPS_DEXT_01061]	Supported scenarios for the definition of *access permission*

[TPS_DEXT_01062]	Existence of DiagnosticServiceClass.accessPermissionValidity in an incomplete model
[TPS_DEXT_01063]	Existence of DiagnosticServiceClass.accessPermissionValidity in a complete model
[TPS_DEXT_01064]	Textually formulated content attached to DiagnosticTroubleCode
[TPS_DEXT_01065]	Different approaches to provide semi-formal textual content attached to a DiagnosticTroubleCode
[TPS_DEXT_01066]	Standardized values of DiagnosticTroubleCode.introduction.trace
[TPS_DEXT_01067]	Textually formulated content attached to DiagnosticEvent
[TPS_DEXT_01068]	Textual description with respect to the DiagnosticEvent
[TPS_DEXT_01069]	Standardized values of DiagnosticEvent.introduction.structuredReq
[TPS_DEXT_01070]	Description of textually semi-formal formulated pre- and post-conditions for the validity of DiagnosticAccessPermission
[TPS_DEXT_01071]	Standardized values of DiagnosticAccessPermission.introduction.structuredReq
[TPS_DEXT_01072]	Purpose of attribute DiagnosticDataIdentifier.representsVin
[TPS_DEXT_01073]	Diagnostic properties that are specific to an individual EcuInstance
[TPS_DEXT_01074]	Difference between the attributes DiagnosticComControl.specificChannel and DiagnosticComControl.subNodeChannel
[TPS_DEXT_01075]	standardized values for the attribute DiagnosticControlDTCSetting.category
[TPS_DEXT_01076]	Identification of sub-functions of diagnostic service ControlDTCSetting
[TPS_DEXT_01077]	Modeling of DiagnosticRoutine
[TPS_DEXT_01078]	Not possible to use the attribute category for the identification of the sub-function of diagnostic service RoutineControl
[TPS_DEXT_01079]	Modeling of the arguments to a DiagnosticRoutine
[TPS_DEXT_01080]	Diagnostic Routine needs to be started
[TPS_DEXT_01081]	Modeling of DiagnosticSessionControl
[TPS_DEXT_01082]	Existence of DiagnosticSessionControl.diagnosticSession
[TPS_DEXT_01083]	Semantics of a DiagnosticEvent
[TPS_DEXT_01084]	DiagnosticEvent can be connected to one or multiple indicators
[TPS_DEXT_01085]	Semantics of DiagnosticConditionGroups
[TPS_DEXT_01086]	Reference to DiagnosticOperationCycle
[TPS_DEXT_01087]	Semantics of DiagnosticOperationCycle
[TPS_DEXT_01088]	Semantics of DiagnosticRoutine.id
[TPS_DEXT_01089]	Definition of an *identifier* of a DiagnosticIOControl
[TPS_DEXT_01090]	Diagnostic service RequestFileTransfer does not define any sub-functions
[TPS_DEXT_03000]	ISO 14229-1 reserves values of DiagnosticTroubleCodeGroup.groupNumber
[TPS_DEXT_03001]	Different types of conditions
[TPS_DEXT_03002]	Two kind of mappings
[TPS_DEXT_03003]	Semantics of DiagnosticEventToTroubleCodeUdsMapping
[TPS_DEXT_03004]	DiagnosticEvent and DiagnosticDebounceAlgorithmProps
[TPS_DEXT_03005]	Existence of DiagnosticEventToDebounceAlgorithmMapping
[TPS_DEXT_03006]	Values of the individual DiagnosticStorageConditions
[TPS_DEXT_03007]	Semantics of DiagnosticEventPortMapping
[TPS_DEXT_03008]	Semantics of DiagnosticExtendedDataRecord
[TPS_DEXT_03009]	Semantics of DiagnosticFreezeFrame
[TPS_DEXT_03010]	Combination of DiagnosticConditions to DiagnosticConditionGroups
[TPS_DEXT_03011]	Clearing request for a DiagnosticEvent

[TPS_DEXT_03012]	Three kinds of DTCs
[TPS_DEXT_03013]	Common properties of a DTC
[TPS_DEXT_03014]	Semantics of DiagnosticTroubleCodeGroup
[TPS_DEXT_03015]	EnableConditions have to be put into a DiagnosticEnableConditionGroup
[TPS_DEXT_03016]	StorageConditions have to be put into a DiagnosticStorageConditionGroup
[TPS_DEXT_03017]	Semantics of DiagnosticOperationCyclePortMapping
[TPS_DEXT_03018]	Semantics of DiagnosticEnableConditionPortMapping
[TPS_DEXT_03019]	Semantics of DiagnosticStorageConditionPortMapping
[TPS_DEXT_03020]	Semantics of DiagnosticDemProvidedDataMapping
[TPS_DEXT_03021]	Aging
[TPS_DEXT_03022]	Different kinds of DiagnosticIndicators

Table B.1: Added Specification Items in 4.0.3

B.1.2 Added Constraints in R4.2.1

Number	Heading
[constr_1324]	Existence of attribute DiagnosticDataIdentifier.representsVin
[constr_1325]	Allowed attributes of SwDataDefProps for DiagnosticDataElement.swDataDefProps
[constr_1326]	Existence of a variable-sized array
[constr_1327]	Multiplicity of DiagnosticContributionSet.ecuInstance
[constr_1328]	Consistency of DiagnosticContributionSet.ecuInstance and DiagnosticServiceTable.ecuInstance
[constr_1329]	Existence of concrete sub-classes of DiagnosticServiceClass in the context created by a DiagnosticContributionSet
[constr_1330]	Custom service identifier shall not overlap with standardized service identifiers
[constr_1331]	Existence of DiagnosticEcuReset.customSubFunctionNumber
[constr_1332]	Value range for DiagnosticEcuReset.customSubfunctionNumber
[constr_1333]	Existence of DiagnosticMemoryIdentifier.memoryLowAddress and DiagnosticMemoryIdentifier.memoryHighAddress
[constr_1334]	Existence of DiagnosticComControl.customSubFunctionNumber
[constr_1335]	Possible values for DiagnosticComControl.customSubFunctionNumber
[constr_1336]	Applicable value range for DiagnosticComControlSpecificChannel.subnetNumber
[constr_1337]	Allowed value range for attribute DiagnosticComControlSubNodeChannel.subNodeNumber
[constr_1338]	Maximum number of aggregated DiagnosticReadDataByPeriodicIDClass.periodicRate
[constr_1339]	Existence of DiagnosticRoutine.start
[constr_1340]	Consistency of DiagnosticServiceSwMapping with respect to synchronously called DiagnosticRoutines
[constr_1341]	Consistency of DiagnosticServiceSwMapping with respect to asynchronously called DiagnosticRoutines
[constr_1342]	Possible values for DiagnosticSecurityAccess.requestSeedId
[constr_1343]	Simultaneous existence of the attributes DiagnosticServiceDataMapping.diagnosticDataElement and DiagnosticDataByIdentifier.dataIdentifier

[constr_1344]	Condition for the identification of data types of attributes DiagnosticServiceDataMapping.mappedDataElement and DiagnosticServiceDataMapping.diagnosticDataElement
[constr_1345]	DiagnosticDataElement shall not (finally) be aggregated by a DiagnosticRoutine
[constr_1346]	Allowed values of DiagnosticServiceSwMapping.serviceInstance
[constr_1347]	Existence of attributes of DiagnosticServiceSwMapping
[constr_1349]	Value of udsDtcValue shall be unique
[constr_1350]	Value of DiagnosticTroubleCodeGroup.groupNumber shall be unique
[constr_1351]	Value of DiagnosticTroubleCodeGroup.groupNumber
[constr_1352]	Existence of maxNumberFreezeFrameRecords vs. freezeFrame
[constr_1353]	Applicability of [constr_1352]
[constr_1354]	Existence of attribute DiagnosticTroubleCodeProps.freezeFrameContent
[constr_1355]	Value of recordNumber
[constr_1356]	Value of recordNumber shall be unique
[constr_1357]	Value of recordNumber
[constr_1358]	Value of recordNumber shall be unique
[constr_1359]	Existence of attribute DiagnosticDebounceAlgorithmProps.debounceCounterStorage
[constr_1360]	Usage of DiagEventDebounceMonitorInternal is not supported in the context of DiagnosticDebounceAlgorithmProps
[constr_1361]	Number of DiagnosticEventToEnableConditionGroupMapping elements per DiagnosticEvent
[constr_1362]	Number of DiagnosticEventToStorageConditionGroupMapping elements per DiagnosticEvent
[constr_1365]	Multiplicity of DiagnosticResponseOnEvent.event
[constr_1366]	Event ID in the context of diagnostic service ResponseOnEvent shall be unique
[constr_1376]	Multiplicity of reference DiagnosticTroubleCodeProps.memoryDestination
[constr_1377]	Existence of reference DiagnosticTroubleCodeProps.memoryDestination
[constr_1378]	Value of DiagnosticMemoryDestinationUserDefined.memoryId
[constr_1379]	Existence of DiagnosticMemoryDestinationPrimary
[constr_1380]	Existence of DiagnosticMemoryDestinationMirror

Table B.2: Added Constraints in R4.2.1

B.2 Constraint History of this Document according to AUTOSAR R4.2.2

B.2.1 Added Traceables in 4.2.2

none

B.2.2 Changed Traceables in 4.2.2

none

B.2.3 Deleted Traceables in 4.2.2

none

B.2.4 Added Constraints in 4.2.2

Id	Heading
[constr_1394]	Value of DiagnosticDataElement.maxNumberOfElements depending on its existence

Table B.3: Added Constraints in 4.2.2

B.2.5 Changed Constraints in 4.2.2

none

B.2.6 Deleted Constraints in 4.2.2

none

C Glossary

Artifact This is a Work Product Definition that provides a description and definition for tangible work product types. Artifacts may be composed of other artifacts ([18]).

At a high level, an artifact is represented as a single conceptual file.

AUTOSAR Tool This is a software tool which supports one or more tasks defined as AUTOSAR tasks in the methodology. Depending on the supported tasks, an AUTOSAR tool can act as an authoring tool, a converter tool, a processor tool or as a combination of those (see separate definitions).

AUTOSAR Authoring Tool An AUTOSAR Tool used to create and modify AUTOSAR XML Descriptions. Example: System Description Editor.

AUTOSAR Converter Tool An AUTOSAR Tool used to create AUTOSAR XML files by converting information from other AUTOSAR XML files. Example: ECU Flattener

AUTOSAR Definition This is the definition of parameters which can have values. One could say that the parameter values are Instances of the definitions. But in the meta model hierarchy of AUTOSAR, definitions are also instances of the meta model and therefore considered as a description. Examples for AUTOSAR definitions are: `EcucParameterDef`, `PostBuildVariantCriterion`, `SwSystemconst`.

AUTOSAR XML Description In AUTOSAR this means "filled Template". In fact an AUTOSAR XML description is the XML representation of an AUTOSAR model.

The AUTOSAR XML description can consist of several files. Each individual file represents an AUTOSAR partial model and shall validate successfully against the AUTOSAR XML schema.

AUTOSAR Meta-Model This is an UML2.0 model that defines the language for describing AUTOSAR systems. The AUTOSAR meta-model is an UML representation of the AUTOSAR templates. UML2.0 class diagrams are used to describe the attributes and their interrelationships. Stereotypes, UML tags and OCL expressions (object constraint language) are used for defining specific semantics and constraints.

AUTOSAR Model This is a representation of an AUTOSAR product. The AUTOSAR model represents aspects suitable to the intended use according to the AUTOSAR methodology.

Strictly speaking, this is an instance of the AUTOSAR meta-model. The information contained in the AUTOSAR model can be anything that is representable according to the AUTOSAR meta-model.

AUTOSAR Partial Model In AUTOSAR, the possible partitioning of models is marked in the meta-model by ≪`atpSplitable`≫. One partial model is represented in an AUTOSAR XML description by one file. The partial model does not need to fulfill all semantic constraints applicable to an AUTOSAR model.

AUTOSAR Processor Tool An AUTOSAR Tool used to create non-AUTOSAR files by processing information from AUTOSAR XML files. Example: RTE Generator

AUTOSAR Template The term "Template" is used in AUTOSAR to describe the format different kinds of descriptions. The term template comes from the idea, that AUTOSAR defines a kind of form which shall be filled out in order to describe a model. The filled form is then called the description.

In fact the AUTOSAR templates are now defined as a meta model.

AUTOSAR XML Schema This is a W3C XML schema that defines the language for exchanging AUTOSAR models. This Schema is derived from the AUTOSAR meta model. The AUTOSAR XML Schema defines the AUTOSAR data exchange format.

Blueprint This is a model from which other models can be derived by copy and refinement. Note that in contrast to meta model resp. types, this process is *not* an instantiation.

Instance Generally this is a particular exemplar of a model or of a type.

Life Cycle Life Cycle is the course of development/evolutionary stages of a model element during its life time.

Meta-Model This defines the building blocks of a model. In that sense, a Meta-Model represents the language for building models.

Meta-Data This includes pertinent information about data, including information about the authorship, versioning, access-rights, timestamps etc.

Model A Model is an simplified representation of reality. The model represents the aspects suitable for an intended purpose.

Partial Model This is a part of a model which is intended to be persisted in one particular artifact.

Pattern in GST : This is an approach to simplify the definition of the meta model by applying a model transformation. This transformation creates an enhanced model out of an annotated model.

Property A property is a structural feature of an object. As an example a “connector” has the properties “receive port” and “send port”

Properties are made variant by the ≪`atpVariation`≫.

Prototype This is the implementation of a role of a type within the definition of another type. In other words a type may contain Prototypes that in turn are typed by "Types". Each one of these prototypes becomes an instance when this type is instantiated.

Type A type provides features that can appear in various roles of this type.

Value This is a particular value assigned to a “Definition”.

Variability Variability of a system is its quality to describe a set of variants. These variants are characterized by variant specific property settings and / or selections. As an example, such a system property selection manifests itself in a particular “receive port” for a connection.

This is implemented using the ≪atpVariation≫.

Variant A system variant is a concrete realization of a system, so that all its properties have been set respectively selected. The software system has no variability anymore with respect to the binding time.

This is implemented using `EvaluatedVariantSet`.

Variation Binding A variant is the result of a variation binding process that resolves the variability of the system by assigning particular values/selections to all the system’s properties.

This is implemented by `VariationPoint`.

Variation Binding Time The variation binding time determines the step in the methodology at which the variability given by a set of variable properties is resolved.

This is implemented by `vh.LatestBindingtime` at the related properties .

Variation Definition Time The variation definition time determines the step in the methodology at which the variation points are defined.

Variation Point A variation point indicates that a property is subject to variation. Furthermore, it is associated with a condition and a binding time which define the system context for the selection / setting of a concrete variant.

This is implemented by `VariationPoint`.

D Modeling of InstanceRef

D.1 Introduction

The existence of so-called InstanceRefs is a direct consequence to the usage of the type-prototype pattern for modeling within AUTOSAR. When referencing a prototype it is also necessary to include a reference to the prototypes typed by their corresponding types that in turn aggregate further prototypes to set up the context.

In other words, InstanceRefs are representing **structured references** that, on the one hand, consist of references to context prototypes (indicated by a subsetting or redefinition of atpContextElement) and finally a reference to the applicable target prototype (indicated by a redefinition of atpTarget).

Note that it is not uncommon to have more than a single context in the modeling of particular InstanceRefs.

For the reader of specifications, the modeling of InstanceRefs manifests as a UML dependency stereotyped ≪instanceRef≫ drawn from one meta-class to another.

This is a simplified indication that the source of the dependency implements an InstanceRef to the meta-class at the target of the dependency. Again, in most cases this is everything a reader needs to understand in order to figure out the modeling.

The formal modeling of InstanceRefs is done by creating subclasses of the abstract meta-class AtpInstanceRef.

Wherever a more detailed understanding of the modeling is advised in the context of the specific chapter of this document, the modeling of a specific subclasses of AtpInstanceRef is explained directly in the context of the corresponding chapter.

In all other cases, a deeper understanding of the modeling of particular subclasses of AtpInstanceRefs can be obtained from reading this chapter.

Class tables included in this chapter are not fully filled out in the sense that most of the notes inside the class tables are missing.

The **primary** purpose of these class tables is to **provide information about the intended order** in which InstanceRefs are **serialized in M1 AUTOSAR models**.

In particular, the information about the order in serialized M1 models can be obtained from the value of the tag xml.sequenceOffset of each attribute of an InstanceRef meta-class.

For more information about the general concept of modeling AtpInstanceRef (e.g. the conceptual background of redefining or subsetting an association from a subclass of AtpInstanceRef to other meta-classes) please refer to [19].

D.2 Modeling

Class	**DataPrototypeInSystemInstanceRef**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::InstanceRefs			
Note				
Base	ARObject,AtpInstanceRef			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
base	System	1	ref	This represents the base of the InstanceRef **Stereotypes:** atpDerived **Tags:** xml.sequenceOffset=10
contextComponent	SwComponentPrototype	*	ref	**Tags:** xml.sequenceOffset=30
contextDataPrototype (ordered)	ApplicationCompositeElementDataPrototype	*	ref	**Tags:** xml.sequenceOffset=50
contextPort	PortPrototype	1	ref	This represents the PortPrototype that is contained in the InstanceRef. **Tags:** xml.sequenceOffset=40
contextRootComposition	RootSwCompositionPrototype	0..1	ref	**Tags:** xml.sequenceOffset=20
targetDataPrototype	DataPrototype	1	ref	This represents the target of the InstanceRef **Tags:** xml.sequenceOffset=60

Table D.1: DataPrototypeInSystemInstanceRef

Class	**SwcServiceDependencyInCompositionInstanceRef**			
Package	M2::AUTOSARTemplates::DiagnosticExtract::InstanceRefs			
Note				
Base	ARObject,AtpInstanceRef			
Attribute	***Datatype***	***Mul.***	***Kind***	***Note***
base	CompositionSwComponentType	1	ref	**Tags:** xml.sequenceOffset=10
contextSwComponentPrototype	SwComponentPrototype	1..*	ref	**Tags:** xml.sequenceOffset=30
rootContext	RootSwCompositionPrototype	0..1	ref	This identifies the rootSoftwareComposition if the instanceRef is modelled in the System context. **Tags:** xml.sequenceOffset=20
targetSwcServiceDependency	SwcServiceDependency	1	ref	**Tags:** xml.sequenceOffset=40

Table D.2: SwcServiceDependencyInCompositionInstanceRef

E Upstream Mapping

E.1 Introduction

This chapter describes the mapping of the ECU Configuration parameters (M1 model) onto the meta-classes and attributes of the AUTOSAR upstream templates (System Template, SW Component Template and ECU Resource Template).

The relationships between upstream templates and ECU Configuration are described in order to answer typical questions like:

- How shall a supplier use the information in a System Description in order to fulfill the needs defined by the systems engineer?
- How is a tool vendor supposed to generate an ECU Configuration Description out of ECU Extract of System Description?

Please note that the tables contain the following columns:

bsw module: Name of BSW module

bsw context: Reference to parameter container

bsw type: Type of parameter

bsw param: Name of the BSW parameter

bsw desc: Description from the configuration document

m2 template: System Template, SW Component Template, ECU Resource Template

m2 param: Name of the upstream template parameter

m2 description: Description from the upstream template definition

mapping rule: Textual description on how to transform between M2 and BSW domains

mapping type:

- local: no mapping needed since parameter local to BSW
- partial: some data can be automatically mapped but not all
- full: all data can be automatically mapped

E.2 Dcm

BSW Module	BSW Context	
Dcm	Dcm	
BSW Parameter		**BSW Type**
DcmConfigSet		EcucParamConfContainerDef

BSW Description	
This container contains the configuration parameters and sub containers of the DCM module supporting multiple configuration sets.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet	
BSW Parameter		**BSW Type**
DcmDsd		EcucParamConfContainerDef
BSW Description		
These parameters configure the Diagnostic Service Dispatcher submodule.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsd	
BSW Parameter		**BSW Type**
DcmDsdRequestManufacturerNotificationEnabled		EcucBooleanParamDef
BSW Description		
Allows to enable or disable the requested notification mechanism for the Manufacturer.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
		local
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsd	
BSW Parameter		**BSW Type**
DcmDsdRequestSupplierNotificationEnabled		EcucBooleanParamDef
BSW Description		
Allows to enable or disable the requested notification mechanism for the Supplier.		
Template Description		

M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsd
BSW Parameter	**BSW Type**
DcmDsdServiceRequestManufacturerNotification	EcucParamConfContainerDef
BSW Description	
The name of this container is used to define the name of the R-Port through which the DCM accesses the interface ServiceRequestNotification. The R-Port is named ServiceRequestManufacturerNotification_{Name} where {Name} is the name of the container DcmDsdServiceRequestManufacturerNotification. The lowerMultiplicity is 0: If DcmDsdRequestManufacturerNotificationEnabled = false the Indication API is not available.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsd
BSW Parameter	**BSW Type**
DcmDsdServiceRequestSupplierNotification	EcucParamConfContainerDef
BSW Description	
The name of this container is used to define the name of the R-Port through which the DCM accesses the interface ServiceRequestNotification. The R-Port is named ServiceRequestSupplierNotification_<SWC> where <SWC> is the name of the container DcmDsdServiceRequestSupplierNotification. The lowerMultiplicity is 0: If DcmDsdRequestSupplierNotification = false the Indication API is not available.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsd

BSW Parameter	BSW Type
DcmDsdServiceTable	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (DSD parameters) for a Service Identifier Table. Note: It is allowed to add OBD services to a DcmDsdServiceTable related to a UDS Protocol. But it is not allowed to add UDS services to a DcmDsdServiceTable related to an OBD Protocol.	
Template Description	
M2 Parameter	

Mapping Rule	Mapping Type
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsd/DcmDsdServiceTable
BSW Parameter	**BSW Type**
DcmDsdService	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (DSD parameters) for a Service.	
Template Description	
M2 Parameter	

Mapping Rule	Mapping Type
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsd/DcmDsdServiceTable/DcmDsdService
BSW Parameter	**BSW Type**
DcmDsdServiceUsed	EcucBooleanParamDef
BSW Description	
Allows to activate or deactivate the usage of a Service. This parameter can be used for multi-purpose ECUs. true - service is available false - service is not available	
Template Description	
M2 Parameter	

Mapping Rule	Mapping Type
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsd/DcmDsdServiceTable/DcmDsdService	
BSW Parameter		**BSW Type**
DcmDsdSidTabFnc		EcucFunctionNameDef
BSW Description		
Callback function of the ECU Supplier specific component for the particular service. The function's prototype is as described for <Module>_<DiagnosticService>. If this parameter is not configured, the service is handled Dcm-internally.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
		local
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsd/DcmDsdServiceTable/DcmDsdService	
BSW Parameter		**BSW Type**
DcmDsdSidTabModeRuleRef		EcucReferenceDef
BSW Description		
Reference to a DcmDspModeRule which controls the execution of the service. If there is no reference configured, no mode rule check shall be performed.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
		local
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsd/DcmDsdServiceTable/DcmDsdService	
BSW Parameter		**BSW Type**
DcmDsdSidTabSecurityLevelRef		EcucReferenceDef
BSW Description		
Reference to a Security Level in which the service is allowed to be executed. Multiple references are allowed for a service. Please refer to ISO 14229-1, ISO 15031-5 and chapter "Verification of the Service Security Access levels." If there is no reference configured, no service security verification shall be performed.		
Template Description		
This represents the associated DiagnosticSecurityLevels		
M2 Parameter		
DiagnosticExtract::Dcm::DiagnosticAccessPermission.securityLevel		
Mapping Rule		**Mapping Type**
1:1 mapping		full

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsd/DcmDsdServiceTable/DcmDsdService
BSW Parameter	**BSW Type**
DcmDsdSidTabServiceId	EcucIntegerParamDef
BSW Description	
Identifier of the service. The possible service identifiers are defined in ISO 14229-1 and ISO 15031-5.	
Template Description	
This meta-class provides the ability to define common properties that are shared among all instances of sub-classes of DiagnosticServiceInstance.	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticService::CommonService::DiagnosticServiceClass	
Mapping Rule	**Mapping Type**
Service identifiers of the used DiagnosticServiceClass	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsd/DcmDsdServiceTable/DcmDsdService
BSW Parameter	**BSW Type**
DcmDsdSidTabSessionLevelRef	EcucReferenceDef
BSW Description	
Reference to a Session Level in which the service is allowed to be executed. Multiple references are allowed for a service. Please refer to ISO 14229-1, ISO 15031-5 and chapter "Verification of the Diagnostic Session". If there is no reference configured, no diagnostic session verification shall be performed.	
Template Description	
This represents the associated DiagnosticSessions	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticAccessPermission.diagnosticSession	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsd/DcmDsdServiceTable/DcmDsdService
BSW Parameter	**BSW Type**
DcmDsdSidTabSubfuncAvail	EcucBooleanParamDef
BSW Description	

Information about whether the service has subfunctions or not. This parameter is used for the handling of the "suppressPosRspMsgIndicationBit" as defined in ISO 14229-1, which can be used as a reference for the configuration. true - service has subfunctions, suppressPosRspMsgIndicationBit is available false - service has no subfunctions, suppressPosRspMsgIndicationBit is not available
Template Description
The category is a keyword that specializes the semantics of the Identifiable. It affects the expected existence of attributes and the applicability of constraints.
M2 Parameter
GenericStructure::GeneralTemplateClasses::Identifiable::Identifiable.category

Mapping Rule	**Mapping Type**
Stanadardized sub-functions of diagnostic services are mainly identified by the category. There are further specific attributes in the meta-model that allow for handling custom subfunctions,	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsd/DcmDsdServiceTable/DcmDsdService

BSW Parameter	**BSW Type**
DcmDsdSubService	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (DSD parameters) for a subservice of a service. Only those services may have subservices, which have the DcmDsdSidTabSubfuncAvail configured as TRUE and have no DcmDsdSidTabFnc configured.	
Template Description	
M2 Parameter	

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsd/DcmDsdServiceTable/DcmDsdService/DcmDsd SubService

BSW Parameter	**BSW Type**
DcmDsdSubServiceFnc	EcucFunctionNameDef
BSW Description	
Callback function of the ECU Supplier specific component for the particular service. The function's prototype is as described for <Module>_<DiagnosticService>_<SubService>. If this parameter is not configured, the subservice is handled Dcm-internally.	
Template Description	
M2 Parameter	

Mapping Rule	**Mapping Type**
	local

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsd/DcmDsdServiceTable/DcmDsdService/DcmDsdSubService
BSW Parameter	**BSW Type**
DcmDsdSubServiceId	EcucIntegerParamDef
BSW Description	
Identifier of the subservice. The possible subservice identifiers are defined in ISO 14229-1 and ISO 15031-5.	
Template Description	
The category is a keyword that specializes the semantics of the Identifiable. It affects the expected existence of attributes and the applicability of constraints.	
M2 Parameter	
GenericStructure::GeneralTemplateClasses::Identifiable::Identifiable.category	
Mapping Rule	**Mapping Type**
Numerical values of diagnostic service according to ISO 14229 correspond to values of DiagnosticServiceInstance.category.	partial
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsd/DcmDsdServiceTable/DcmDsdService/DcmDsdSubService
BSW Parameter	**BSW Type**
DcmDsdSubServiceModeRuleRef	EcucReferenceDef
BSW Description	
Reference to a DcmDspModeRule which controls the execution of the subservice. If there is no reference configured, no mode rule check shall be performed.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsd/DcmDsdServiceTable/DcmDsdService/DcmDsdSubService
BSW Parameter	**BSW Type**
DcmDsdSubServiceSecurityLevelRef	EcucReferenceDef
BSW Description	

Reference to a Security Level in which the subservice is allowed to be executed. Multiple references are allowed for a subservice. Please refer to ISO 14229-1, ISO 15031-5 and chapter "Verification of the Service Security Access levels." If there is no reference configured, no subservice security verification shall be performed.
Template Description
This represents the associated DiagnosticSecurityLevels
M2 Parameter
DiagnosticExtract::Dcm::DiagnosticAccessPermission.securityLevel

Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsd/DcmDsdServiceTable/DcmDsdService/DcmDsdSubService

BSW Parameter	**BSW Type**
DcmDsdSubServiceSessionLevelRef	EcucReferenceDef

BSW Description
Reference to a Session Level in which the subservice is allowed to be executed. Multiple references are allowed for a subservice. Please refer to ISO 14229-1, ISO 15031-5 and chapter "Verification of the Diagnostic Session". If there is no reference configured, no diagnostic session verification shall be performed.
Template Description
This represents the associated DiagnosticSessions
M2 Parameter
DiagnosticExtract::Dcm::DiagnosticAccessPermission.diagnosticSession

Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsd/DcmDsdServiceTable/DcmDsdService/DcmDsdSubService

BSW Parameter	**BSW Type**
DcmDsdSubServiceUsed	EcucBooleanParamDef

BSW Description
Allows to activate or deactivate the usage of a Subservice. This parameter can be used for multi-purpose ECUs. true - subservice is available false - subservice is not available.
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsd/DcmDsdServiceTable
BSW Parameter	**BSW Type**
DcmDsdSidTabId	EcucIntegerParamDef
BSW Description	
Due to using possibly more service tables, the unique DcmDsdSidTabId can be used to identify them.	
Template Description	
M2 Parameter	

Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet
BSW Parameter	**BSW Type**
DcmDsl	EcucParamConfContainerDef
BSW Description	
These parameters configure the Diagnostic Session Layer submodule.	
Template Description	
M2 Parameter	

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsl
BSW Parameter	**BSW Type**
DcmDslBuffer	EcucParamConfContainerDef
BSW Description	
This container contains the configuration of a diagnostic buffer.	
Template Description	
M2 Parameter	

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslBuffer
BSW Parameter	**BSW Type**
DcmDslBufferSize	EcucIntegerParamDef
BSW Description	
Size of the diagnostic buffer in bytes. For a linear buffer the size shall be as large as the longest diagnostic message (request or response). For a paged buffer the size has impacts on the application performance.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsl
BSW Parameter	**BSW Type**
DcmDslCallbackDCMRequestService	EcucParamConfContainerDef
BSW Description	
Each DcmDslCallbackDCMRequestService container defines an R-Port with the CallbackDCMRequestServices interface which the Dcm uses to ask permission for protocol changes from the application software. The R-Port has the name CallbackDCMRequestServices_<SWC> where <SWC> is the name of this container.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsl
BSW Parameter	**BSW Type**
DcmDslDiagResp	EcucParamConfContainerDef
BSW Description	
This container contains the configuration of the automatic requestCorrectlyReceivedResponsePending response management in the Dcm.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslDiagResp
BSW Parameter	**BSW Type**
DcmDslDiagRespMaxNumRespPend	EcucIntegerParamDef
BSW Description	
Maximum number of negative responses with response code 0x78 (requestCorrectlyReceivedResponsePending) allowed for a request. If Dcm reaches this limit, an automatic 0x10 (generalReject) final response will be trasmitted and the service processing will be cancelled.	
Template Description	
Maximum number of negative responses with response code 0x78 (requestCorrectlyReceivedResponsePending) allowed per request. DCM will send a negative response with response code 0x10 (generalReject), in case the limit value gets reached. Value 0xFF means that no limit number of NRC 0x78 response apply.	
M2 Parameter	
DiagnosticExtract::DiagnosticCommonProps::DiagnosticCommonProps.maxNumberOfRequestCorrectlyReceivedResponsePending	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslDiagResp
BSW Parameter	**BSW Type**
DcmDslDiagRespOnSecondDeclinedRequest	EcucBooleanParamDef
BSW Description	
Defines the reaction upon a second request (ClientB) that can not be processed (e.g. due to priority assessment). TRUE: when the second request (Client B) can not be processed, it shall be answered with NRC21 BusyRepeatRequest. FALSE: when the second request (Client B) can not be processed, it shall not be responded.	
Template Description	
Defines the reaction upon a second request (ClientB) that can not be processed (e.g. due to priority assessment). TRUE: when the second request (Client B) can not be processed, it shall be answered with NRC21 BusyRepeatRequest. FALSE: when the second request (Client B) can not be processed, it shall not be responded.	
M2 Parameter	
DiagnosticExtract::DiagnosticCommonProps::DiagnosticCommonProps.responseOnSecondDeclinedRequest	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsl	
BSW Parameter		**BSW Type**
DcmDslProtocol		EcucParamConfContainerDef
BSW Description		
This container contains the configurations of the diagnostic protocols used in Dcm.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol	
BSW Parameter		**BSW Type**
DcmDslProtocolRow		EcucParamConfContainerDef
BSW Description		
This container contains the configuration of one particular diagnostic protocol used in Dcm.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow	
BSW Parameter		**BSW Type**
DcmDslConnection		EcucChoiceContainerDef
BSW Description		
This container contains the configuration of a communication channel for one particular protocol. Note that it is allowed to communicate with multiple testers, therefore multiple connections may be configured for a protocol.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context

Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow/DcmDslConnection

BSW Parameter	BSW Type
DcmDslMainConnection	EcucParamConfContainerDef

BSW Description

This container contains the configuration for a main connection of a diagnostic protocol. Additionally it may contain references to ROE and Periodic connections if the protocol type or protocol transmission type needs them.

Template Description

M2 Parameter

Mapping Rule	Mapping Type
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow/DcmDslConnection/DcmDslMainConnection

BSW Parameter	BSW Type
DcmDslPeriodicTransmissionConRef	EcucReferenceDef

BSW Description

Reference to a periodic transmission connection which is used for the processing of periodic transmission events.

Template Description

M2 Parameter

Mapping Rule	Mapping Type
Auto-generation	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow/DcmDslConnection/DcmDslMainConnection

BSW Parameter	BSW Type
DcmDslProtocolComMChannelRef	EcucSymbolicNameReferenceDef

BSW Description

Reference to the ComMChannel on which the DcmDslProtocolRxPdu is received and the DcmDslProtocolTxPdu is transmitted.

Template Description

The CommunicationCluster is the main element to describe the topological connection of communicating ECUs.

A cluster describes the ensemble of ECUs, which are linked by a communication medium of arbitrary topology (bus, star, ring, ...). The nodes within the cluster share the same communication protocol, which may be event-triggered, time-triggered or a combination of both.

A CommunicationCluster aggregates one or more physical channels.

M2 Parameter

SystemTemplate::Fibex::FibexCore::CoreTopology::CommunicationCluster	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow/DcmDsl Connection/DcmDslMainConnection
BSW Parameter	**BSW Type**
DcmDslProtocolRx	EcucParamConfContainerDef
BSW Description	
This container contains the configuration parameters of a reception channel in a diagnostic connection. Note that the combination of a DcmDspProtocolRxPduId and a DcmDspProtocolRxAddrType shall be unique for each configured reception channel. Also note that only one channel with DcmDspProtocolRxAddrType == DCM_PHYSICAL_TYPE is allowed for a connection.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow/DcmDsl Connection/DcmDslMainConnection/DcmDslProtocolRx
BSW Parameter	**BSW Type**
DcmDslProtocolRxAddrType	EcucEnumerationParamDef
BSW Description	
Selects the addressing type of the reception channel. Physical addressing is used for 1:1 communication, functional addressing is used for 1:N communication. For details refer to ISO 14229-1.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**

Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow/DcmDslConnection/DcmDslMainConnection/DcmDslProtocolRx/DcmDslProtocolRxAddrType

BSW Parameter	**BSW Type**
DCM_FUNCTIONAL_TYPE	EcucEnumerationLiteralDef
BSW Description	
FUNCTIONAL = 1 to n communication	
Template Description	
Reference to functional request messages.	
M2 Parameter	
SystemTemplate::DiagnosticConnection::DiagnosticConnection.functionalRequest	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow/DcmDslConnection/DcmDslMainConnection/DcmDslProtocolRx/DcmDslProtocolRxAddrType

BSW Parameter	**BSW Type**
DCM_PHYSICAL_TYPE	EcucEnumerationLiteralDef
BSW Description	
PHYSICAL = 1 to 1 communications using physical addressing	
Template Description	
Reference to a physical request message.	
M2 Parameter	
SystemTemplate::DiagnosticConnection::DiagnosticConnection.physicalRequest	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow/DcmDslConnection/DcmDslMainConnection/DcmDslProtocolRx

BSW Parameter	**BSW Type**
DcmDslProtocolRxPduId	EcucIntegerParamDef
BSW Description	
Identifier of the PDU that is used for this reception channel.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Auto-geernation	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**

Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow/DcmDslConnection/DcmDslMainConnection/DcmDslProtocolRx

BSW Parameter	BSW Type
DcmDslProtocolRxPduRef	EcucReferenceDef

BSW Description

Reference to a Pdu in EcuC that is used for this reception channel.

Template Description

The IPdu (Interaction Layer Protocol Data Unit) element is used to sum up all Pdus that are routed by the PduR.

M2 Parameter

SystemTemplate::Fibex::FibexCore::CoreCommunication::IPdu

Mapping Rule	Mapping Type
Reference to IPdu of xxxTpConnection for DiagnosticConnection.physicalRequest / DiagnosticConnection.functionalRequest	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow/DcmDslConnection/DcmDslMainConnection

BSW Parameter	BSW Type
DcmDslProtocolRxTesterSourceAddr	EcucIntegerParamDef

BSW Description

Source address of the tester which uses this connection for diagnostic communication.

The parameter is not required for generic connections, where the MetaDataLength of a PDU is greater than or equal to 1.

Template Description

An ECU specific ID for responses of diagnostic routines.

M2 Parameter

SystemTemplate::Fibex::FibexCore::CoreTopology::EcuInstance.diagnosticAddress

Mapping Rule	Mapping Type
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow/DcmDslConnection/DcmDslMainConnection

BSW Parameter	BSW Type
DcmDslProtocolTx	EcucParamConfContainerDef

BSW Description

This container contains the configuration parameters of a transmission channel in a diagnostic connection.

Template Description

In the vast majority of cases a response is required. However, there are also cases where providing the response is not possible and/or not allowed.

M2 Parameter

SystemTemplate::DiagnosticConnection::DiagnosticConnection.response

Mapping Rule	Mapping Type
1:1 mapping	full
Mapping Status	**Mapping ID**

valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow/DcmDslConnection/DcmDslMainConnection/DcmDslProtocolTx

BSW Parameter	BSW Type
DcmDslProtocolTxPduRef	EcucReferenceDef

BSW Description
Reference to a Pdu in EcuC that is used for this transmission channel.

Template Description
The IPdu (Interaction Layer Protocol Data Unit) element is used to sum up all Pdus that are routed by the PduR.

M2 Parameter
SystemTemplate::Fibex::FibexCore::CoreCommunication::IPdu

Mapping Rule	Mapping Type
Reference to IPdu of xxxTpConnection for DiagnosticConnection.response	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow/DcmDslConnection/DcmDslMainConnection/DcmDslProtocolTx

BSW Parameter	BSW Type
DcmDslTxConfirmationPduId	EcucIntegerParamDef

BSW Description
Identifier of the PDU that is used by the lower level module for transmission confirmation of responses on this channel.

Template Description

M2 Parameter

Mapping Rule	Mapping Type
Auto-generation	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow/DcmDslConnection/DcmDslMainConnection

BSW Parameter	BSW Type
DcmDslROEConnectionRef	EcucReferenceDef

BSW Description
Reference to a ResponseOnEvent connection which is used for the processing of ResponseOnEvent events.

Template Description

M2 Parameter

Mapping Rule	Mapping Type
Auto-generation	local

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow/DcmDslConnection
BSW Parameter	**BSW Type**
DcmDslPeriodicTransmission	EcucParamConfContainerDef
BSW Description	
This container contains the configuration of a periodic transmission connection.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow/DcmDslConnection/DcmDslPeriodicTransmission
BSW Parameter	**BSW Type**
DcmDslPeriodicConnection	EcucParamConfContainerDef
BSW Description	
This container contains the configuration of a transmission channel for a periodic transmission connection.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow/DcmDslConnection/DcmDslPeriodicTransmission/DcmDslPeriodicConnection
BSW Parameter	**BSW Type**
DcmDslPeriodicTxConfirmationPduId	EcucIntegerParamDef
BSW Description	
Identifier of the PDU that is used by the lower level module for transmission confirmation of responses on this channel.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Auto-generation	local

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow/DcmDslConnection/DcmDslPeriodicTransmission/DcmDslPeriodicConnection
BSW Parameter	**BSW Type**
DcmDslPeriodicTxPduRef	EcucReferenceDef
BSW Description	
Reference to a Pdu in EcuC that is used for this periodic transmission channel.	
Template Description	
The IPdu (Interaction Layer Protocol Data Unit) element is used to sum up all Pdus that are routed by the PduR.	
M2 Parameter	
SystemTemplate::Fibex::FibexCore::CoreCommunication::IPdu	
Mapping Rule	**Mapping Type**
Reference to IPdu of xxxTpConnection in case of DiagnosticConnection.periodicResponseTp or IPdu of PduTriggering in case of DiagnosticConnection.periodicResponseUudt	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow/DcmDslConnection
BSW Parameter	**BSW Type**
DcmDslResponseOnEvent	EcucParamConfContainerDef
BSW Description	
This container contains the configuration of a ResponseOnEvent connection.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow/DcmDslConnection/DcmDslResponseOnEvent
BSW Parameter	**BSW Type**
DcmDslRoeTxConfirmationPduId	EcucIntegerParamDef
BSW Description	
Identifier of the PDU that is used by the lower level module for transmission confirmation of responses on this connection.	
Template Description	
M2 Parameter	

Mapping Rule	Mapping Type
Auto-generation	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow/DcmDslConnection/DcmDslResponseOnEvent
BSW Parameter	**BSW Type**
DcmDslRoeTxPduRef	EcucReferenceDef
BSW Description	
Reference to a Pdu in EcuC that is used for this ResponseOnEvent transmission connection.	
Template Description	
The IPdu (Interaction Layer Protocol Data Unit) element is used to sum up all Pdus that are routed by the PduR.	
M2 Parameter	
SystemTemplate::Fibex::FibexCore::CoreCommunication::IPdu	
Mapping Rule	**Mapping Type**
Reference to IPdu of xxxTpConnection for DiagnosticConnection.responseOnEvent	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow
BSW Parameter	**BSW Type**
DcmDslProtocolID	EcucEnumerationParamDef
BSW Description	
The diagnostic protocol type for the DCM DSL protocol that is being configured. Implementation Type: Dcm_ProtocolType	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow/DcmDslProtocolID
BSW Parameter	**BSW Type**
DCM_PERIODICTRANS_ON_CAN	EcucEnumerationLiteralDef
BSW Description	
DCM_PERIODICTRANS_ON_CAN	
Template Description	

AbstractCanPhysicalChannel: Abstract class that is used to collect the common TtCAN and CAN PhysicalChannel attributes. **DiagnosticConnection.periodicResponseUudt**: Reference to UUDT responses. **CanTpConnection**: A connection identifies the sender and the receiver of this particular communication. The CanTp module routes a Pdu through this connection. atpVariation: Derived, because TpNode can vary.
M2 Parameter
SystemTemplate::Fibex::Fibex4Can::CanTopology::AbstractCanPhysicalChannel, SystemTemplate::DiagnosticConnection::DiagnosticConnection.periodicResponseUudt, SystemTemplate::TransportProtocols::CanTpConnection

Mapping Rule	Mapping Type
In case DiagnosticConnection.periodicResponseTp exists and TpConnection Ident.ident belongs to a CanTpConnection or DiagnosticConnection.periodicResponseUudt exists and PhysicalChannel given as AbstractCanPhysicalChannel	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow/DcmDsl ProtocolID

BSW Parameter	BSW Type
DCM_PERIODICTRANS_ON_FLEXRAY	EcucEnumerationLiteralDef
BSW Description	
DCM_PERIODICTRANS_ON_FLEXRAY	
Template Description	
FlexrayTpConnection: A connection identifies the sender and the receiver of this particular communication. The FlexRayTp module routes a Pdu through this connection. In a System Description the references to the PduPools are mandatory. In an ECU Extract these references can be optional: On unicast connections these references are always mandatory. On multicast the txPduPool is mandatory on the sender side. The rxPduPool is mandatory on the receiver side. On Gateway ECUs both references are mandatory. **DiagnosticConnection.periodicResponseUudt**: Reference to UUDT responses. **FlexrayPhysicalChannel**: FlexRay specific attributes to the physicalChannel	
M2 Parameter	
SystemTemplate::TransportProtocols::FlexrayTpConnection, SystemTemplate::DiagnosticConnection::DiagnosticConnection.periodicResponseUudt, SystemTemplate::Fibex::Fibex4Flexray::FlexrayTopology::FlexrayPhysicalChannel	

Mapping Rule	Mapping Type

In case DiagnosticConnection.periodicResponseTp exists and TpConnection Ident.ident belongs to FlexRayTpConnection or DiagnosticConnection.periodicResponseUudt exists and PhysicalChannel given as FlexrayPhysicalChannel	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow/DcmDsl ProtocolID
BSW Parameter	**BSW Type**
DCM_PERIODICTRANS_ON_IP	EcucEnumerationLiteralDef
BSW Description	
DCM_PERIODICTRANS_ON_IP	
Template Description	
SocketConnection: The SoAd serves as a (De)Multiplexer between different PDU sources and the TCP/IP stack. **DiagnosticConnection.periodicResponseUudt**: Reference to UUDT responses. **EthernetPhysicalChannel**: The EthernetPhysicalChannel represents a VLAN or an untagged channel. An untagged channel is modeled as an EthernetPhysicalChannel without an aggregated VLAN.	
M2 Parameter	
SystemTemplate::Fibex::Fibex4Ethernet::EthernetCommunication::SocketConnection, SystemTemplate::DiagnosticConnection::DiagnosticConnection.periodicResponseUudt, SystemTemplate::Fibex::Fibex4Ethernet::EthernetTopology::EthernetPhysicalChannel	
Mapping Rule	**Mapping Type**
In case DiagnosticConnection.periodicResponseTp exists and TpConnection Ident.ident belongs to SocketConnection or DiagnosticConnection.periodicResponseUudt exists and PhysicalChannel given as EthernetPhysicalChannel	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow/DcmDsl ProtocolID
BSW Parameter	**BSW Type**
DCM_ROE_ON_CAN	EcucEnumerationLiteralDef
BSW Description	
DCM_ROE_ON_CAN	
Template Description	

DiagnosticConnection.responseOnEvent: Reference to a ROE message. **FlexrayTpConnection**: A connection identifies the sender and the receiver of this particular communication. The FlexRayTp module routes a Pdu through this connection. In a System Description the references to the PduPools are mandatory. In an ECU Extract these references can be optional: On unicast connections these references are always mandatory. On multicast the txPduPool is mandatory on the sender side. The rxPduPool is mandatory on the receiver side. On Gateway ECUs both references are mandatory.
M2 Parameter
SystemTemplate::DiagnosticConnection::DiagnosticConnection.responseOnEvent, SystemTemplate::TransportProtocols::FlexrayTpConnection

Mapping Rule	**Mapping Type**
In case DiagnosticConnection.responseOnEvent exists and TpConnection Ident.ident belongs to a CanTpConnection	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow/DcmDsl ProtocolID

BSW Parameter	**BSW Type**
DCM_ROE_ON_FLEXRAY	EcucEnumerationLiteralDef
BSW Description	
DCM_ROE_ON_FLEXRAY	
Template Description	
DiagnosticConnection.responseOnEvent: Reference to a ROE message. **FlexrayTpConnection**: A connection identifies the sender and the receiver of this particular communication. The FlexRayTp module routes a Pdu through this connection. In a System Description the references to the PduPools are mandatory. In an ECU Extract these references can be optional: On unicast connections these references are always mandatory. On multicast the txPduPool is mandatory on the sender side. The rxPduPool is mandatory on the receiver side. On Gateway ECUs both references are mandatory.	
M2 Parameter	
SystemTemplate::DiagnosticConnection::DiagnosticConnection.responseOnEvent, SystemTemplate::TransportProtocols::FlexrayTpConnection	

Mapping Rule	**Mapping Type**
In case DiagnosticConnection.responseOnEvent exists and TpConnection Ident.ident belongs to FlexRayTpConnection	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**

Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow/DcmDslProtocolID

BSW Parameter	BSW Type
DCM_ROE_ON_IP	EcucEnumerationLiteralDef

BSW Description
DCM_ROE_ON_IP

Template Description
DiagnosticConnection.responseOnEvent: Reference to a ROE message. **SocketConnection**: The SoAd serves as a (De)Multiplexer between different PDU sources and the TCP/IP stack.

M2 Parameter
SystemTemplate::DiagnosticConnection::DiagnosticConnection.responseOnEvent, SystemTemplate::Fibex::Fibex4Ethernet::EthernetCommunication::SocketConnection

Mapping Rule	Mapping Type
In case DiagnosticConnection.responseOnEvent exists and TpConnectionIdent.ident belongs to SocketConnection	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow/DcmDslProtocolID

BSW Parameter	BSW Type
DCM_UDS_ON_CAN	EcucEnumerationLiteralDef

BSW Description
UDS on CAN (ISO15765-3; ISO14229-1)

Template Description
DiagnosticConnection.physicalRequest: Reference to a physical request message. **CanTpConnection**: A connection identifies the sender and the receiver of this particular communication. The CanTp module routes a Pdu through this connection. atpVariation: Derived, because TpNode can vary.

M2 Parameter
SystemTemplate::DiagnosticConnection::DiagnosticConnection.physicalRequest, SystemTemplate::TransportProtocols::CanTpConnection

Mapping Rule	Mapping Type
In case DiagnosticConnection.physicalRequest exists and TpConnectionIdent.ident belongs to CanTpConnection	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow/DcmDslProtocolID

BSW Parameter	BSW Type
DCM_UDS_ON_FLEXRAY	EcucEnumerationLiteralDef

BSW Description

DCM_UDS_ON_FLEXRAY UDS on FlexRay (Manufacturer specific; ISO14229-1)	
Template Description	
DiagnosticConnection.physicalRequest: Reference to a physical request message. **FlexrayTpConnection**: A connection identifies the sender and the receiver of this particular communication. The FlexRayTp module routes a Pdu through this connection. In a System Description the references to the PduPools are mandatory. In an ECU Extract these references can be optional: On unicast connections these references are always mandatory. On multicast the txPduPool is mandatory on the sender side. The rxPduPool is mandatory on the receiver side. On Gateway ECUs both references are mandatory.	
M2 Parameter	
SystemTemplate::DiagnosticConnection::DiagnosticConnection.physicalRequest, SystemTemplate::TransportProtocols::FlexrayTpConnection	
Mapping Rule	**Mapping Type**
In case DiagnosticConnection.physicalRequest exists and TpConnection Ident.ident belongs to FlexRayTpConnection	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow/DcmDsl ProtocolID
BSW Parameter	**BSW Type**
DCM_UDS_ON_IP	EcucEnumerationLiteralDef
BSW Description	
DCM_UDS_ON_IP	
Template Description	
DiagnosticConnection.physicalRequest: Reference to a physical request message. **SocketConnection**: The SoAd serves as a (De)Multiplexer between different PDU sources and the TCP/IP stack.	
M2 Parameter	
SystemTemplate::DiagnosticConnection::DiagnosticConnection.physicalRequest, SystemTemplate::Fibex::Fibex4Ethernet::EthernetCommunication::SocketConnection	
Mapping Rule	**Mapping Type**
In case DiagnosticConnection.physicalRequest exists and TpConnection Ident.ident belongs to a SocketConnection	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow
BSW Parameter	**BSW Type**
DcmDslProtocolMaximumResponseSize	EcucIntegerParamDef
BSW Description	
This parameter is mandatory and defines the maximum length of the response message in case DcmPagedBufferEnabled == TRUE	

Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow
BSW Parameter	**BSW Type**
DcmDslProtocolPreemptTimeout	EcucFloatParamDef
BSW Description	
This parameter is the timeout value used in protocol preemption if this protocol preempts another diagnostic protocol. The protocol shall be started maximum DcmDslProtocolPreemptTimeout time after the first request in the new protocol.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow
BSW Parameter	**BSW Type**
DcmDslProtocolPriority	EcucIntegerParamDef
BSW Description	
Protocol priority used during protocol preemption. A higher priority protocol may preempt a lower priority protocol. Lower numeric values represent higher protocol priority: 0 - Highest protocol priority 255 - Lowest protocol priority	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow
BSW Parameter	**BSW Type**

DcmDslProtocolRowUsed	EcucBooleanParamDef
BSW Description	
Allows to activate or deactivate the usage of a Protocol. This parameter can be used for multi-purpose ECUs. true - protocol is available false - protocol is not available.	
Template Description	
M2 Parameter	

Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow
BSW Parameter	**BSW Type**
DcmDslProtocolRxBufferRef	EcucReferenceDef
BSW Description	
Reference to a configured diagnostic buffer that is used for diagnostic request reception for the protocol.	
Template Description	
M2 Parameter	

Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow
BSW Parameter	**BSW Type**
DcmDslProtocolSIDTable	EcucReferenceDef
BSW Description	
Reference to a service table that is used for diagnostic request processing for this protocol.	
Template Description	
M2 Parameter	

Mapping Rule	**Mapping Type**
Auto-generation	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow

BSW Parameter	BSW Type
DcmDslProtocolTransType	EcucEnumerationParamDef
BSW Description	
This parameter is used only if the protocol is of type DCM_ROE_ON_xxx. It selects the transmission type of the protocol.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow/DcmDsl ProtocolTransType
BSW Parameter	**BSW Type**
TYPE1	EcucEnumerationLiteralDef
BSW Description	
Messages on the DcmTxPduId already used for normal diagnostic responses. The outgoing messages must be synchronized with 'normal outgoing messages', which have a higher priority.	
Template Description	
Reference to a ROE message.	
M2 Parameter	
SystemTemplate::DiagnosticConnection::DiagnosticConnection.responseOnEvent	
Mapping Rule	**Mapping Type**
TYPE1 : periodicResponseTp / responseOnEvent using same reference as the normal response	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow/DcmDsl ProtocolTransType
BSW Parameter	**BSW Type**
TYPE2	EcucEnumerationLiteralDef
BSW Description	
Messages on a separate DcmTxPduId.	
Template Description	
Reference to a ROE message.	
M2 Parameter	
SystemTemplate::DiagnosticConnection::DiagnosticConnection.responseOnEvent	
Mapping Rule	**Mapping Type**
TYPE2: periodicResponseTp / responseOnEvent using other reference as the normal response	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context

Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow	
BSW Parameter		**BSW Type**
DcmDslProtocolTxBufferRef		EcucReferenceDef
BSW Description		
Reference to a configured diagnostic buffer that is used for diagnostic response transmission for the protocol.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
		local
Mapping Status		**Mapping ID**
valid		

BSW Module	**BSW Context**	
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow	
BSW Parameter		**BSW Type**
DcmSendRespPendOnTransToBoot		EcucBooleanParamDef
BSW Description		
Parameter specifying if the ECU should send a NRC 0x78 (response pending) before transitioning to the bootloader (parameter set to TRUE) or if the transition shall be initiated without sending NRC 0x78 (parameter set to FALSE).		
Template Description		
The purpose of this attribute is to define whether or not the ECU should send a NRC 0x78 (response pending) before transitioning to the bootloader (in this case the attribute shall be set to "true") or if the transition shall be initiated without sending NRC 0x78 (in this case the attribute shall be set to "false").		
M2 Parameter		
DiagnosticExtract::DiagnosticCommonProps::DiagnosticEcuProps.sendRespPendOnTransToBoot		
Mapping Rule		**Mapping Type**
1:1 mapping		full
Mapping Status		**Mapping ID**
valid		

BSW Module	**BSW Context**	
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow	
BSW Parameter		**BSW Type**
DcmTimStrP2ServerAdjust		EcucFloatParamDef
BSW Description		
This parameter is used to guarantee that the diagnostic response is available on the bus before reaching P2 by adjusting the current DcmDspSessionP2ServerMax. This parameter mainly represents the software architecture dependent communication delay between the time the transmission is initiated by DCM and the time when the message is actually transmitted to the bus. The parameter value is defined in seconds and must be a multiple of DcmTaskTime.		
Template Description		
M2 Parameter		

Mapping Rule	Mapping Type
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsl/DcmDslProtocol/DcmDslProtocolRow
BSW Parameter	**BSW Type**
DcmTimStrP2StarServerAdjust	EcucFloatParamDef
BSW Description	
This parameter is used to guarantee that the diagnostic response is available on the bus before reaching P2Star by adjusting the current DcmDspSessionP2StarServerMax. This parameter mainly represents the software architecture dependent communication delay between the time the transmission is initiated by DCM and the time when the message is actually transmitted to the bus. The parameter value is defined in seconds and must be a multiple of DcmTaskTime.	
Template Description	
M2 Parameter	

Mapping Rule	Mapping Type
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet
BSW Parameter	**BSW Type**
DcmDsp	EcucParamConfContainerDef
BSW Description	
These parameters apply to Diagnostic Service Processing. There will always be one set of these parameters per Dcm.	
Template Description	
M2 Parameter	

Mapping Rule	Mapping Type
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp
BSW Parameter	**BSW Type**
DcmDspClearDTC	EcucParamConfContainerDef
BSW Description	
This container contains the configuration for the Clear DTC service.	
Template Description	

M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspClearDTC
BSW Parameter	**BSW Type**
DcmDspClearDTCCheckFnc	EcucFunctionNameDef
BSW Description	
Callback function for condition check, manufacturer / supplier specific checks on the groupOfDTC, which is requested to clear. This parameter is related to the interface : Xxx_ClearDTCCheckFnc.	
Template Description	
M2 Parameter	

Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspClearDTC
BSW Parameter	**BSW Type**
DcmDspClearDTCModeRuleRef	EcucReferenceDef
BSW Description	
Reference to DcmModeRule Mode rule which controls to clear the DTCs. If there is no reference, no check of the mode rule shall be done.	
Template Description	
M2 Parameter	

Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp
BSW Parameter	**BSW Type**
DcmDspComControl	EcucParamConfContainerDef
BSW Description	

Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspComControl
BSW Parameter	**BSW Type**
DcmDspComControlAllChannel	EcucParamConfContainerDef
BSW Description	
Collection of ComM channels which shall be controlled if all networks are addressed.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspComControl/DcmDspComControlAll Channel
BSW Parameter	**BSW Type**
DcmDspAllComMChannelRef	EcucSymbolicNameReferenceDef
BSW Description	
Reference to ComM channel.	
Template Description	
This reference represents the semantics that all available channels shall be affected. It is still necessary to refer to individual CommunicatuionClusters because there could be private CommunicationClusters in the System Extract that are not subject to the service "communication control". By referring to the applicable CommunicationClusters it can be made sure that only the affected CommunicationClusters are accessed.	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticService::CommunicationControl::DiagnosticComControl Class.allChannels	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspComControl/DcmDspComControlAll Channel
BSW Parameter	**BSW Type**

DcmDspComControlAllChannelUsed	EcucBooleanParamDef
BSW Description	
Allow to activate or deactivate the usage of a ComM channel collection to be controlled, for multi purpose ECUs true = ComM channel collection used false = ComM channel collection not used	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspComControl
BSW Parameter	**BSW Type**
DcmDspComControlSetting	EcucParamConfContainerDef
BSW Description	
Provide the configuration of the Communication control.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspComControl/DcmDspComControlSetting
BSW Parameter	**BSW Type**
DcmDspComControlCommunicationReEnableModeRuleRef	EcucReferenceDef
BSW Description	
Reference to DcmModeRule Mode rule which controls re-enabling of communication by DCM. [ref. SWS_Dcm_00753]	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspComControl

BSW Parameter	BSW Type
DcmDspComControlSpecificChannel	EcucParamConfContainerDef
BSW Description	
Assigns subnet number to ComM channel which will be controlled.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspComControl/DcmDspComControlSpecificChannel
BSW Parameter	**BSW Type**
DcmDspComControlSpecificChannelUsed	EcucBooleanParamDef
BSW Description	
Allow to activate or deactivate the usage of a Subnet assigned to the ComM channel which will be controlled, for multi purpose ECUs. true = Subnet used false = Subnet not used	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspComControl/DcmDspComControlSpecificChannel
BSW Parameter	**BSW Type**
DcmDspSpecificComMChannelRef	EcucSymbolicNameReferenceDef
BSW Description	
Reference to ComM channel.	
Template Description	
This represents the ability to add additional attributes to the case that only specific channels are supposed to be considered,	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticService::CommunicationControl::DiagnosticComControlClass.specificChannel	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspComControl/DcmDspComControlSpecificChannel	
BSW Parameter		**BSW Type**
DcmDspSubnetNumber		EcucIntegerParamDef
BSW Description		
Subnet Number which controls the specific ComMChannel.		
Template Description		
This represents the applicable subnet number (which is an arbitrary number ranging from 1..14)		
M2 Parameter		
DiagnosticExtract::Dcm::DiagnosticService::CommunicationControl::DiagnosticComControlSpecificChannel.subnetNumber		
Mapping Rule		**Mapping Type**
1:1 mapping		full
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspComControl	
BSW Parameter		**BSW Type**
DcmDspComControlSubNode		EcucParamConfContainerDef
BSW Description		
This container gives information about the node identification number and the ComM channel used to address a sub-network.		
Template Description		
M2 Parameter		
DiagnosticExtract::Dcm::DiagnosticService::CommunicationControl::DiagnosticComControlClass.subNodeChannel		
Mapping Rule		**Mapping Type**
1:1 mapping		full
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspComControl/DcmDspComControlSubNode	
BSW Parameter		**BSW Type**
DcmDspComControlSubNodeComMChannelRef		EcucSymbolicNameReferenceDef
BSW Description		
This parameter references a ComM channel where this node is connected to.		
Template Description		
This represents the affected CommunicationClusters in the role subNodeChannel		
M2 Parameter		
DiagnosticExtract::Dcm::DiagnosticService::CommunicationControl::DiagnosticComControlSubNodeChannel.subNodeChannel		
Mapping Rule		**Mapping Type**
1:1 mapping		full
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspComControl/DcmDspComControlSubNode

BSW Parameter	BSW Type
DcmDspComControlSubNodeId	EcucIntegerParamDef

BSW Description
The node identification number DcmDspComControlSubNodeId is addressed by the CommunicationControl (0x28) request.

Template Description
This represents the applicable subNode number. The value corresponds to the request message parameter nodeIdentificationNumber of diagnostic service CommunicationControl (0x28).

M2 Parameter
DiagnosticExtract::Dcm::DiagnosticService::CommunicationControl::DiagnosticComControlSubNodeChannel.subNodeNumber

Mapping Rule	Mapping Type
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspComControl/DcmDspComControlSubNode

BSW Parameter	BSW Type
DcmDspComControlSubNodeUsed	EcucBooleanParamDef

BSW Description
This parameter determines if a node control function is available or not.

Template Description

M2 Parameter

Mapping Rule	Mapping Type
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp

BSW Parameter	BSW Type
DcmDspCommonAuthorization	EcucParamConfContainerDef

BSW Description
This container contains the configuration (parameters) for the common Authorization being equal for several services / sub-services.

Template Description

M2 Parameter

Mapping Rule	Mapping Type
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspCommonAuthorization
BSW Parameter	**BSW Type**
DcmDspCommonAuthorizationModeRuleRef	EcucReferenceDef
BSW Description	
Reference to DcmModeRule Mode rule which controls this service/ sub-service. If there is no reference, no check of the mode rule shall be done.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspCommonAuthorization
BSW Parameter	**BSW Type**
DcmDspCommonAuthorizationSecurityLevelRef	EcucReferenceDef
BSW Description	
Reference to DcmDspSecurityRow Security levels allowed to control this service/ sub-service. If there is no reference, no check of security level shall be done.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspCommonAuthorization
BSW Parameter	**BSW Type**
DcmDspCommonAuthorizationSessionRef	EcucReferenceDef
BSW Description	
Reference to DcmDspSessionRow Sessions allowed to control this service/ sub-service. If there is no reference, no check of session level shall be done.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp	
BSW Parameter		**BSW Type**
DcmDspControlDTCSetting		EcucParamConfContainerDef
BSW Description		
Provide the configuration of the ControlDTCSetting mechanism.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspControlDTCSetting	
BSW Parameter		**BSW Type**
DcmDspControlDTCSettingReEnableModeRuleRef		EcucReferenceDef
BSW Description		
Reference to DcmModeRule Mode rule which controls re-enabling of controlDTCsetting by DCM. The DCM module shall execute a ControlDTCSetting.Off (call Dem_DcmEnableDTCSetting()) in case that the referenced mode rule is not fulfilled anymore.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
		local
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspControlDTCSetting	
BSW Parameter		**BSW Type**
DcmSupportDTCSettingControlOptionRecord		EcucBooleanParamDef
BSW Description		
This configuration switch defines if the DTCSettingControlOptionRecord is in general supported in the request message or not.		
Template Description		
This represents the decision whether the DTCSettingControlOptionRecord (see ISO 14229-1) is in general supported in the request message.		
M2 Parameter		
DiagnosticExtract::Dcm::DiagnosticService::ControlDTCSetting::DiagnosticControlDTCSetting Class.controlOptionRecordPresent		
Mapping Rule		**Mapping Type**
1:1 mapping		full
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp	
BSW Parameter		**BSW Type**
DcmDspDDDIDcheckPerSourceDID		EcucBooleanParamDef
BSW Description		
Defines the check for session, security and mode dependencies per source DIDs with a ReadDataByIdentifier (0x22). true: Dcm module shall check the session, security and mode dependencies per source DIDs with a ReadDataByIdentifier (0x22) with DID in the range 0xF200 to 0xF3FF false: Dcm module shall not check the session, security and mode dependencies per source DIDs with a ReadDataByIdentifier (0x22) with DID in the range 0xF200 to 0xF3FF		
Template Description		
If set to TRUE, the Dcm module shall check the session, security and mode dependencies per source DIDs with a ReadDataByIdentifier (0x22) with DID in the range 0xF200 to 0xF3FF. If set to FALSE. the Dcm module shall not check the session, security and mode dependencies per source DIDs with a ReadDataByIdentifier (0x22) with DID in the range 0xF200 to 0xF3FF.		
M2 Parameter		
DiagnosticExtract::Dcm::DiagnosticService::DynamicallyDefineDataIdentifier::DiagnosticDynamicallyDefineDataIdentifierClass.checkPerSourceId		
Mapping Rule		**Mapping Type**
1:1 mapping		full
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp	
BSW Parameter		**BSW Type**
DcmDspData		EcucParamConfContainerDef
BSW Description		
This container contains the configuration (parameters) of a Data belonging to a DID		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData	
BSW Parameter		**BSW Type**
DcmDspDataBlockIdRef		EcucSymbolicNameReferenceDef
BSW Description		
NRAM blockId to access the data. Only relevant if DcmDspDataUsePort==USE_BLOCK_ID.		
Template Description		

M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData
BSW Parameter	**BSW Type**
DcmDspDataConditionCheckReadFnc	EcucFunctionNameDef
BSW Description	
Function name to demand application if the conditions (e.g. System state) to read the DID are correct. (ConditionCheckRead-function). Multiplicity shall be equal to parameter DcmDspDataReadFnc. Only relevant if * DcmDspDataConditionCheckReadFncUsed is set to 'TRUE' and * DcmDspDataUsePort=="USE_DATA_SYNCH_FNC or * DcmDspDataUsePort==USE_DATA_ASYNCH_FNC" or * DcmDspDataUsePort==USE_DATA_ASYNCH_FNC_ERROR". This parameter is related to the interface Xxx_ConditionCheckRead.	
Template Description	
This is supposed to represent a reference to a BswServiceDependency. the latter is not derived from Referrable and therefore this detour needs to be implemented to still let BswServiceDependency become the target of a reference.	
M2 Parameter	
DiagnosticExtract::ServiceMapping::DiagnosticServiceSwMapping.mappedBswServiceDependency	
Mapping Rule	**Mapping Type**
It could be possible to get the FNC name via BswServiceDependency	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData
BSW Parameter	**BSW Type**
DcmDspDataConditionCheckReadFncUsed	EcucBooleanParamDef
BSW Description	

This parameter determines if a condition check function is available or not. If the parameter is set to 'TRUE' and DcmDspDataUsePort is set to * 'USE_DATA_ASYNCH_CLIENT_SERVER' or * 'USE_DATA_ASYNCH_CLIENT_SERVER_ERROR' or * 'USE_DATA_SYNCH_CLIENT_SERVER', the DCM shall generate the according function call. If the parameter is set to 'TRUE' and DcmDspDataUsePort is set to * 'USE_DATA_SYNCH_FNC' or * 'USE_DATA_ASYNCH_FNC_ERROR' * 'USE_DATA_ASYNCH_FNC', the parameter 'DcmDspDataConditionCheckReadFnc' shall contain a valid C-function.
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData

BSW Parameter	**BSW Type**
DcmDspDataEcuSignal	EcucFunctionNameDef
BSW Description	
Function name to control the access to a certain ECU Signal by the DCM. (IoHwAb_Dcm_<symbolic name of ECU signal>-function). Only relevant if DcmDspDataUsePort==USE_ECU_SIGNAL.	
Template Description	
M2 Parameter	

Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData

BSW Parameter	**BSW Type**
DcmDspDataEndianness	EcucEnumerationParamDef
BSW Description	
Defines the endianness of the data belonging to a DID in a diagnostic request or response message. If no DcmDspDataEndiness is defined the value of DcmDspDataDefaultEndianness is applicable.	
Template Description	
This attribute specifies the byte order of the base type.	

M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.byteOrder	
Mapping Rule	**Mapping Type**
baseType.baseTypeDefinition.byteOrder referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticDataIdentifier.dataElement	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData
BSW Parameter	**BSW Type**
DcmDspDataFreezeCurrentStateFnc	EcucFunctionNameDef
BSW Description	
Function name to request to application to freeze the current state of an IOControl. (FreezeCurrentState-function). Only relevant if * DcmDspDataUsePort=="USE_DATA_SYNCH_FNC or * DcmDspDataUsePort==USE_DATA_ASYNCH_FNC" or * DcmDspDataUsePort==USE_DATA_ASYNCH_FNC_ERROR". This parameter is related to the interface Xxx_FreezeCurrentState.	
Template Description	
This is supposed to represent a reference to a BswServiceDependency. the latter is not derived from Referrable and therefore this detour needs to be implemented to still let BswServiceDependency become the target of a reference.	
M2 Parameter	
DiagnosticExtract::ServiceMapping::DiagnosticServiceSwMapping.mappedBswServiceDependency	
Mapping Rule	**Mapping Type**
It could be possible to get the FNC name via BswServiceDependency	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData
BSW Parameter	**BSW Type**
DcmDspDataGetScalingInfoFnc	EcucFunctionNameDef
BSW Description	
Function name to request to application the scaling information of the DID. (GetScalingInformation-function). Only relevant if * DcmDspDataUsePort=="USE_DATA_SYNCH_FNC or * DcmDspDataUsePort==USE_DATA_ASYNCH_FNC" or * DcmDspDataUsePort==USE_DATA_ASYNCH_FNC_ERROR". This parameter is related to the interface Xxxx_GetScalingInformation.	
Template Description	
M2 Parameter	

Mapping Rule	Mapping Type
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData
BSW Parameter	**BSW Type**
DcmDspDataInfoRef	EcucReferenceDef
BSW Description	
Reference to 1 DcmDspDataInfo	
Template Description	
M2 Parameter	

Mapping Rule	Mapping Type
Auto-generation	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData
BSW Parameter	**BSW Type**
DcmDspDataReadDataLengthFnc	EcucFunctionNameDef
BSW Description	
Function name to request from application the data length of a DID. (ReadDataLength-function). Only relevant if * DcmDspDataUsePort=="USE_DATA_SYNCH_FNC or * DcmDspDataUsePort==USE_DATA_ASYNCH_FNC" or * DcmDspDataUsePort==USE_DATA_ASYNCH_FNC_ERROR". This parameter is related to the interface Xxx_ReadDataLength.	
Template Description	
This is supposed to represent a reference to a BswServiceDependency. the latter is not derived from Referrable and therefore this detour needs to be implemented to still let BswServiceDependency become the target of a reference.	
M2 Parameter	
DiagnosticExtract::ServiceMapping::DiagnosticServiceSwMapping.mappedBswServiceDependency	

Mapping Rule	Mapping Type
It could be possible to get the FNC name via BswServiceDependency	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData
BSW Parameter	**BSW Type**
DcmDspDataReadEcuSignal	EcucFunctionNameDef
BSW Description	

Function name for read access to a certain ECU Signal by the DCM. (IoHwAb_Dcm_Read<EcuSignalName>-function). Only relevant if DcmDspDataUsePort==USE_ECU_SIGNAL.
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData

BSW Parameter	**BSW Type**
DcmDspDataReadFnc	EcucFunctionNameDef

BSW Description
Function name to request from application the data value of a DID. (ReadData-function). Multiplicity shall be equal to parameter DcmDspDataConditionCheckReadFnc. Only relevant if * DcmDspDataConditionCheckReadFncUsed is set to 'TRUE' and * DcmDspDataUsePort=="USE_DATA_SYNCH_FNC or * DcmDspDataUsePort==USE_DATA_ASYNCH_FNC" or * DcmDspDataUsePort==USE_DATA_ASYNCH_FNC_ERROR". This parameter is related to the interface Xxx_ReadData.
Template Description
This is supposed to represent a reference to a BswServiceDependency. the latter is not derived from Referrable and therefore this detour needs to be implemented to still let BswServiceDependency become the target of a reference.
M2 Parameter
DiagnosticExtract::ServiceMapping::DiagnosticServiceSwMapping.mappedBswServiceDependency

Mapping Rule	**Mapping Type**
It could be possible to get the FNC name via BswServiceDependency	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData

BSW Parameter	**BSW Type**
DcmDspDataResetToDefaultFnc	EcucFunctionNameDef
BSW Description	

Function name to request to application to reset an IOControl to default value. (ResetToDefault-function). Only relevant if * DcmDspDataUsePort=="USE_DATA_SYNCH_FNC or * DcmDspDataUsePort==USE_DATA_ASYNCH_FNC" or * DcmDspDataUsePort==USE_DATA_ASYNCH_FNC_ERROR". This parameter is related to the interface Xxx_ResetToDefault.
Template Description
This is supposed to represent a reference to a BswServiceDependency. the latter is not derived from Referrable and therefore this detour needs to be implemented to still let BswServiceDependency become the target of a reference.
M2 Parameter
DiagnosticExtract::ServiceMapping::DiagnosticServiceSwMapping.mappedBswServiceDepen-dency

Mapping Rule	**Mapping Type**
It could be possible to get the FNC name via BswServiceDependency	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData

BSW Parameter	**BSW Type**
DcmDspDataReturnControlToEcuFnc	EcucFunctionNameDef

BSW Description
Function name to request to application to return control to ECU of an IOControl. (ReturnControlToECU-function). Only relevant if * DcmDspDataUsePort=="USE_DATA_SYNCH_FNC or * DcmDspDataUsePort==USE_DATA_ASYNCH_FNC" or * DcmDspDataUsePort==USE_DATA_ASYNCH_FNC_ERROR". This parameter is related to the interface Xxx_ReturnControlToECU.
Template Description
This is supposed to represent a reference to a BswServiceDependency. the latter is not derived from Referrable and therefore this detour needs to be implemented to still let BswServiceDependency become the target of a reference.
M2 Parameter
DiagnosticExtract::ServiceMapping::DiagnosticServiceSwMapping.mappedBswServiceDepen-dency

Mapping Rule	**Mapping Type**
It could be possible to get the FNC name via BswServiceDependency	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData

BSW Parameter	**BSW Type**
DcmDspDataShortTermAdjustmentFnc	EcucFunctionNameDef

BSW Description

Function name to request to application to adjust the IO signal. (ShortTermAdjustment-function). Only relevant if * DcmDspDataUsePort=="USE_DATA_SYNCH_FNC or * DcmDspDataUsePort==USE_DATA_ASYNCH_FNC" or * DcmDspDataUsePort==USE_DATA_ASYNCH_FNC_ERROR". This parameter is related to the interface Xxx_ShortTermAdjustment.
Template Description
This is supposed to represent a reference to a BswServiceDependency. the latter is not derived from Referrable and therefore this detour needs to be implemented to still let BswServiceDependency become the target of a reference.
M2 Parameter
DiagnosticExtract::ServiceMapping::DiagnosticServiceSwMapping.mappedBswServiceDependency

Mapping Rule	**Mapping Type**
It could be possible to get the FNC name via BswServiceDependency	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData

BSW Parameter	**BSW Type**
DcmDspDataSize	EcucIntegerParamDef

BSW Description
Length of data in bits associated to the Data. If Data has variable datalength, that corresponds to the maximum datalength.
Template Description
The existence of this attribute turns the data instance into an array of data. The attribute determines the size of the array in terms of how many elements the array can take.
M2 Parameter
DiagnosticExtract::CommonDiagnostics::DiagnosticDataElement.maxNumberOfElements

Mapping Rule	**Mapping Type**
S/R via array: DcmDspDataSize= maxNumberOfElements*baseTypeSize C/S of FNC callback: DcmDspDataSize= maxNumberOfElements*8 Note: 8 is the baseTypeSize of UINT8	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData

BSW Parameter	**BSW Type**
DcmDspDataType	EcucEnumerationParamDef

BSW Description
Provide the implementation data type of data belonging to a DID.
Template Description
M2 Parameter

Mapping Rule	Mapping Type
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDataType
BSW Parameter	**BSW Type**
BOOLEAN	EcucEnumerationLiteralDef
BSW Description	
Type of the data is boolean.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
baseTypeEncoding = BOOLEAN baseTypeSize = 1 maxNumberOfElements shall not exist arraySizeSemantics shall not exist	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDataType
BSW Parameter	**BSW Type**
SINT16	EcucEnumerationLiteralDef
BSW Description	
Type of the data is sint16.	
Template Description	
BaseTypeDirectDefinition.baseTypeSize: Describes the length of the data type specified in the container in bits. **BaseTypeDirectDefinition.baseTypeEncoding**: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding	
Mapping Rule	**Mapping Type**
baseTypeEncoding = 2C baseTypeSize = 16 maxNumberOfElements shall not exist arraySizeSemantics shall not exist	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDataType	
BSW Parameter		**BSW Type**
SINT16_N		EcucEnumerationLiteralDef
BSW Description		
Type of the data is sint16 array.		
Template Description		
BaseTypeDirectDefinition.baseTypeSize: Describes the length of the data type specified in the container in bits. **BaseTypeDirectDefinition.baseTypeEncoding**: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **DiagnosticDataElement.arraySizeSemantics**: This attribute controls the meaning of the value of the array size. **DiagnosticDataElement.maxNumberOfElements**: The existence of this attribute turns the data instance into an array of data. The attribute determines the size of the array in terms of how many elements the array can take.		
M2 Parameter		
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding DiagnosticExtract::CommonDiagnostics::DiagnosticDataElement.arraySizeSemantics DiagnosticExtract::CommonDiagnostics::DiagnosticDataElement.maxNumberOfElements		
Mapping Rule		**Mapping Type**
baseTypeEncoding = 2C baseTypeSize = 16 maxNumberOfElements exists and value is greater than 0 (cf. TPS_DEXT_01001) arraySizeSemantics either does not exist or exists and is set to ArraySizeSemanticsEnum.fixedSize (cf. TPS_DEXT_01001)		full
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDataType	
BSW Parameter		**BSW Type**
SINT32		EcucEnumerationLiteralDef
BSW Description		
Type of the data is sint32.		
Template Description		
BaseTypeDirectDefinition.baseTypeSize: Describes the length of the data type specified in the container in bits. **BaseTypeDirectDefinition.baseTypeEncoding**: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence.		
M2 Parameter		
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding		
Mapping Rule		**Mapping Type**

baseTypeEncoding = 2C baseTypeSize = 32 maxNumberOfElements shall not exist arraySizeSemantics shall not exist	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDataType
BSW Parameter	**BSW Type**
SINT32_N	EcucEnumerationLiteralDef
BSW Description	
Type of the data is sint32 array.	
Template Description	
BaseTypeDirectDefinition.baseTypeSize: Describes the length of the data type specified in the container in bits. **BaseTypeDirectDefinition.baseTypeEncoding**: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **DiagnosticDataElement.arraySizeSemantics**: This attribute controls the meaning of the value of the array size. **DiagnosticDataElement.maxNumberOfElements**: The existence of this attribute turns the data instance into an array of data. The attribute determines the size of the array in terms of how many elements the array can take.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding DiagnosticExtract::CommonDiagnostics::DiagnosticDataElement.arraySizeSemantics DiagnosticExtract::CommonDiagnostics::DiagnosticDataElement.maxNumberOfElements	
Mapping Rule	**Mapping Type**
baseTypeEncoding = 2C baseTypeSize = 32 maxNumberOfElements exists and value is greater than 0 (cf. TPS_DEXT_01001) arraySizeSemantics either does not exist or exists and is set to ArraySizeSemanticsEnum.fixedSize (cf. TPS_DEXT_01001)	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDataType
BSW Parameter	**BSW Type**
SINT8	EcucEnumerationLiteralDef
BSW Description	
Type of the data is sint8.	
Template Description	

BaseTypeDirectDefinition.baseTypeSize: Describes the length of the data type specified in the container in bits. **BaseTypeDirectDefinition.baseTypeEncoding**: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding	
Mapping Rule	**Mapping Type**
baseTypeEncoding = 2C baseTypeSize = 8 maxNumberOfElements shall not exist arraySizeSemantics shall not exist	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDataType
BSW Parameter	**BSW Type**
SINT8_N	EcucEnumerationLiteralDef
BSW Description	
Type of the data is sint8 array.	
Template Description	
BaseTypeDirectDefinition.baseTypeSize: Describes the length of the data type specified in the container in bits. **BaseTypeDirectDefinition.baseTypeEncoding**: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **DiagnosticDataElement.arraySizeSemantics**: This attribute controls the meaning of the value of the array size. **DiagnosticDataElement.maxNumberOfElements**: The existence of this attribute turns the data instance into an array of data. The attribute determines the size of the array in terms of how many elements the array can take.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding DiagnosticExtract::CommonDiagnostics::DiagnosticDataElement.arraySizeSemantics DiagnosticExtract::CommonDiagnostics::DiagnosticDataElement.maxNumberOfElements	
Mapping Rule	**Mapping Type**
baseTypeEncoding = 2C baseTypeSize = 8 maxNumberOfElements exists and value is greater than 0 (cf. TPS_DEXT_01001) arraySizeSemantics either does not exist or exists and is set to ArraySizeSemanticsEnum.fixedSize (cf. TPS_DEXT_01001)	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDataType	
BSW Parameter		**BSW Type**
UINT16		EcucEnumerationLiteralDef
BSW Description		
Type of the data is uint16.		
Template Description		
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.		
M2 Parameter		
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize		
Mapping Rule		**Mapping Type**
baseTypeEncoding = NONE baseTypeSize = 16 maxNumberOfElements shall not exist arraySizeSemantics shall not exist		full
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDataType	
BSW Parameter		**BSW Type**
UINT16_N		EcucEnumerationLiteralDef
BSW Description		
Type of the data is uint16 array.		
Template Description		
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits. **DiagnosticDataElement.arraySizeSemantics**: This attribute controls the meaning of the value of the array size. **DiagnosticDataElement.maxNumberOfElements**: The existence of this attribute turns the data instance into an array of data. The attribute determines the size of the array in terms of how many elements the array can take.		
M2 Parameter		
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize DiagnosticExtract::CommonDiagnostics::DiagnosticDataElement.arraySizeSemantics DiagnosticExtract::CommonDiagnostics::DiagnosticDataElement.maxNumberOfElements		
Mapping Rule		**Mapping Type**

baseTypeEncoding = NONE baseTypeSize = 16 maxNumberOfElements exists and value is greater than 0 (cf. TPS_DEXT_01001) arraySizeSemantics either does not exist or exists and is set to ArraySizeSemanticsEnum.fixedSize (cf. TPS_DEXT_01001)	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDataType
BSW Parameter	**BSW Type**
UINT32	EcucEnumerationLiteralDef
BSW Description	
Type of the data is uint32.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
baseTypeEncoding = NONE baseTypeSize = 32 maxNumberOfElements shall not exist arraySizeSemantics shall not exist	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDataType
BSW Parameter	**BSW Type**
UINT32_N	EcucEnumerationLiteralDef
BSW Description	
Type of the data is uint32 array.	
Template Description	

BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits. **DiagnosticDataElement.arraySizeSemantics**: This attribute controls the meaning of the value of the array size. **DiagnosticDataElement.maxNumberOfElements**: The existence of this attribute turns the data instance into an array of data. The attribute determines the size of the array in terms of how many elements the array can take.
M2 Parameter
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize DiagnosticExtract::CommonDiagnostics::DiagnosticDataElement.arraySizeSemantics DiagnosticExtract::CommonDiagnostics::DiagnosticDataElement.maxNumberOfElements

Mapping Rule	**Mapping Type**
baseTypeEncoding = NONE baseTypeSize = 32 maxNumberOfElements exists and value is greater than 0 (cf. TPS_DEXT_01001) arraySizeSemantics either does not exist or exists and is set to ArraySizeSemanticsEnum.fixedSize (cf. TPS_DEXT_01001)	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDataType

BSW Parameter	**BSW Type**
UINT8	EcucEnumerationLiteralDef

BSW Description
Type of the data is uint8.
Template Description
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.
M2 Parameter
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize

Mapping Rule	**Mapping Type**
baseTypeEncoding = NONE baseTypeSize = 8 maxNumberOfElements shall not exist arraySizeSemantics shall not exist	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDataType	
BSW Parameter		**BSW Type**
UINT8_DYN		EcucEnumerationLiteralDef
BSW Description		
Type of the data is uint8 array with dynamic length.		
Template Description		
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits. **DiagnosticDataElement.arraySizeSemantics**: This attribute controls the meaning of the value of the array size. **DiagnosticDataElement.maxNumberOfElements**: The existence of this attribute turns the data instance into an array of data. The attribute determines the size of the array in terms of how many elements the array can take.		
M2 Parameter		
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize DiagnosticExtract::CommonDiagnostics::DiagnosticDataElement.arraySizeSemantics DiagnosticExtract::CommonDiagnostics::DiagnosticDataElement.maxNumberOfElements		
Mapping Rule		**Mapping Type**
baseTypeEncoding = NONE baseTypeSize = 8 maxNumberOfElements exists and value is greater than 0 (cf. TPS_DEXT_01002) arraySizeSemantics exists and is set to ArraySizeSemanticsEnum.variableSize (cf. TPS_DEXT_01002)		full
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDataType	
BSW Parameter		**BSW Type**
UINT8_N		EcucEnumerationLiteralDef
BSW Description		
Type of the data is uint8 array.		
Template Description		

BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits. **DiagnosticDataElement.arraySizeSemantics**: This attribute controls the meaning of the value of the array size. **DiagnosticDataElement.maxNumberOfElements**: The existence of this attribute turns the data instance into an array of data. The attribute determines the size of the array in terms of how many elements the array can take.
M2 Parameter
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize DiagnosticExtract::CommonDiagnostics::DiagnosticDataElement.arraySizeSemantics DiagnosticExtract::CommonDiagnostics::DiagnosticDataElement.maxNumberOfElements

Mapping Rule	**Mapping Type**
baseTypeEncoding = NONE baseTypeSize = 8 maxNumberOfElements exists and value is greater than 0 (cf. TPS_DEXT_01001) arraySizeSemantics either does not exist or exists and is set to ArraySizeSemanticsEnum.fixedSize (cf. TPS_DEXT_01001)	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData

BSW Parameter	**BSW Type**
DcmDspDataUsePort	EcucEnumerationParamDef
BSW Description	
Defines which interface shall be used to access the data.	
Template Description	
This attribute controls whether interaction requires the software-component to react synchronously on a request or whether it processes the request in background but still the DCM has to issue the call again to eventually obtain the result of the request.	
M2 Parameter	
CommonStructure::ServiceNeeds::DiagnosticValueNeeds.processingStyle	

Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	up_Dcm_00001

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDataUsePort

BSW Parameter	**BSW Type**
USE_DATA_ASYNCH_CLIENT_SERVER	EcucEnumerationLiteralDef
BSW Description	

The DCM will access the Data using an R-Port requiring a asynchronous ClientServertInterface DataServices_{Data}. The R-Port is named DataServices_{Data} where {Data} is the name of the container DcmDspData.
Template Description
The software-component processes the request in background but still the DCM has to issue the call again to eventually obtain the result of the request.
M2 Parameter
CommonStructure::ServiceNeeds::DiagnosticProcessingStyleEnum.processingStyleAsynchronous

Mapping Rule	**Mapping Type**
DiagnosticServiceSwMapping is having a SwcServiceDependency and ServiceNeeds::DiagnosticProcessingStyleEnum is equal to processingStyleAsynchronous	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDataUsePort

BSW Parameter	**BSW Type**
USE_DATA_ASYNCH_CLIENT_SERVER_ERROR	EcucEnumerationLiteralDef
BSW Description	
The Dcm will access the Data using an R-Port requiring a asynchronous ClientServertInterface DataServices_{Data}. The parameter ErrorCode can be returned to allow the application to trigger a negative response during the operation. The R-Port is named DataServices_{Data} where {Data} is the name of the container DcmDspData.	
Template Description	
The software-component is superposed to react synchronously on the request.	
M2 Parameter	
CommonStructure::ServiceNeeds::DiagnosticProcessingStyleEnum.processingStyleSynchronous	

Mapping Rule	**Mapping Type**
DiagnosticServiceSwMapping is having a SwcServiceDependency and ServiceNeeds::DiagnosticProcessingStyleEnum is equal to processingStyleSynchronous	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDataUsePort

BSW Parameter	**BSW Type**
USE_DATA_ASYNCH_FNC	EcucEnumerationLiteralDef
BSW Description	
The DCM will access the Data using the functions that are defined in parameters of type EcucFunctionNameDef (but without DcmDspDataReadDataLengthFnc) in the DcmDspData container. DCM_E_PENDING return is allowed. OpStatus is existing as IN parameter.	
Template Description	
The software-component processes the request in background but still the DCM has to issue the call again to eventually obtain the result of the request.	
M2 Parameter	
CommonStructure::ServiceNeeds::DiagnosticProcessingStyleEnum.processingStyleAsynchronous	

Mapping Rule	**Mapping Type**
DiagnosticServiceSwMapping is having a BswServiceDependency and ServiceNeeds::DiagnosticProcessingStyleEnum is equal to processingStyleAsynchronous	full

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDataUsePort
BSW Parameter	**BSW Type**
USE_DATA_ASYNCH_FNC_ERROR	EcucEnumerationLiteralDef
BSW Description	
The DCM will access the Data using the functions that are defined in parameters of type EcucFunctionNameDef (but without DcmDspDataReadDataLengthFnc) in the DcmDspData container. DCM_E_PENDING return is allowed. OpStatus is existing as IN parameter. The parameter ErrorCode can be returned to allow the application to trigger a negative response during the operation.	
Template Description	
The software-component is superposed to react synchronously on the request.	
M2 Parameter	
CommonStructure::ServiceNeeds::DiagnosticProcessingStyleEnum.processingStyleSynchronous	
Mapping Rule	**Mapping Type**
DiagnosticServiceSwMapping is having a SwcServiceDependency and ServiceNeeds::DiagnosticProcessingStyleEnum is equal to processingStyleSynchronous	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDataUsePort
BSW Parameter	**BSW Type**
USE_DATA_SENDER_RECEIVER	EcucEnumerationLiteralDef
BSW Description	
The DCM will access the Data using an Port requiring a SenderReceiverInteface (with isService=false) DataServices_{Data}. The Port is namedDataServices_{Data} where {Data} is the name of the container DcmDspData.	
Template Description	
This represents the ability to define a mapping of a diagnostic service to a software-component. This kind of service mapping is applicable for the usage of SenderReceiverInterfaces.	
M2 Parameter	
DiagnosticExtract::ServiceMapping::DiagnosticServiceDataMapping	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDataUsePort
BSW Parameter	**BSW Type**
USE_DATA_SENDER_RECEIVER_AS_SERVICE	EcucEnumerationLiteralDef
BSW Description	
The DCM will access the Data using an service Port requiring a SenderReceiverInteface (with isService=true) DataServices_{Data} . The Port is namedDataServices_{Data} where {Data} is the name of the container DcmDspData.	
Template Description	

This represents the ability to define a mapping of a diagnostic service to a software-component. This kind of service mapping is applicable for the usage of SenderReceiverInterfaces.	
M2 Parameter	
DiagnosticExtract::ServiceMapping::DiagnosticServiceDataMapping	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDataUsePort
BSW Parameter	**BSW Type**
USE_DATA_SYNCH_CLIENT_SERVER	EcucEnumerationLiteralDef
BSW Description	
The DCM will access the Data using an R-Port requiring a synchronous ClientServertInterface DataServices_{Data}. The R-Port is named DataServices_{Data} where {Data} is the name of the container DcmDspData.	
Template Description	
The software-component is superposed to react synchronously on the request.	
M2 Parameter	
CommonStructure::ServiceNeeds::DiagnosticProcessingStyleEnum.processingStyleSynchronous	
Mapping Rule	**Mapping Type**
DiagnosticServiceSwMapping is having a SwcServiceDependency and ServiceNeeds::DiagnosticProcessingStyleEnum is equal to processingStyleSynchronous	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDataUsePort
BSW Parameter	**BSW Type**
USE_DATA_SYNCH_FNC	EcucEnumerationLiteralDef
BSW Description	
The DCM will access the Data using the functions that are defined in parameters of type EcucFunctionNameDef (but without DcmDspDataReadDataLengthFnc) in the DcmDspData container. DCM_E_PENDING return value is not allowed and OpStatus parameter is not existing in the prototype.	
Template Description	
The software-component is superposed to react synchronously on the request.	
M2 Parameter	
CommonStructure::ServiceNeeds::DiagnosticProcessingStyleEnum.processingStyleSynchronous	
Mapping Rule	**Mapping Type**
DiagnosticServiceSwMapping is having a BswServiceDependency and ServiceNeeds::DiagnosticProcessingStyleEnum is equal to processingStyleSynchronous	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData
BSW Parameter	**BSW Type**

DcmDspDataWriteFnc	EcucFunctionNameDef
BSW Description	
Function name to request application to write the data value of a DID. (WriteData-function). Only relevant if * DcmDspDataUsePort=="USE_DATA_SYNCH_FNC or * DcmDspDataUsePort==USE_DATA_ASYNCH_FNC" or * DcmDspDataUsePort==USE_DATA_ASYNCH_FNC_ERROR". This parameter is related to the interface Xxx_WriteData.	
Template Description	
This is supposed to represent a reference to a BswServiceDependency. the latter is not derived from Referrable and therefore this detour needs to be implemented to still let BswServiceDependency become the target of a reference.	
M2 Parameter	
DiagnosticExtract::ServiceMapping::DiagnosticServiceSwMapping.mappedBswServiceDependency	
Mapping Rule	**Mapping Type**
It could be possible to get the FNC name via BswServiceDependency	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData
BSW Parameter	**BSW Type**
DcmDspDiagnosisScaling	EcucChoiceContainerDef
BSW Description	
This container contains the configuration (parameters) of an alternative Diagnosis Representation. Out if this the scaling between Diagnosis and ECU internal representation and vice versa can be calculated.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDiagnosisScaling
BSW Parameter	**BSW Type**
DcmDspAlternativeDataInterface	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) of an alternative Diagnosis Representation by the means of a VariableDataPrototype in a DataInterface. Additionally a reference to PortInterfaceMapping can be defined which provide already the mapping rules between the VariableDataPrototype in a DataInterface used by the software component (DcmDspExternalSRDataElementClass) and the intended Diagnosis Representation defined by DcmDataElement.	
Template Description	

M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDiagnosisScaling/DcmDspAlternativeDataInterface

BSW Parameter	**BSW Type**
DcmDataElement	EcucForeignReferenceDef

BSW Description
Alternative Diagnosis Representation for the data defined by the means of a VariableDataPrototype in a DataInterface.
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDiagnosisScaling/DcmDspAlternativeDataInterface

BSW Parameter	**BSW Type**
DcmPortInterfaceMapping	EcucForeignReferenceDef

BSW Description
Optional reference to PortInterfaceMapping which defines the mapping rules.
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDiagnosisScaling

BSW Parameter	**BSW Type**
DcmDspAlternativeDataProps	EcucParamConfContainerDef

BSW Description

This container contains the configuration (parameters) of an alternative Diagnosis Representation by the means of ECU configuration parameters. The physical unit of the alternative data representation is defined by the DataPrototype referenced by DcmDspExternalSRDataElementClass. Additionally the definition of a text table mapping can be a defined for DcmDspDataTypeCategory TEXTTABLE and SCALE_LINEAR_AND_TEXTTABLE
Template Description
This class is a collection of properties relevant for data objects under various aspects. One could consider this class as a "pattern of inheritance by aggregation". The properties can be applied to all objects of all classes in which SwDataDefProps is aggregated. Note that not all of the attributes or associated elements are useful all of the time. Hence, the process definition (e.g. expressed with an OCL or a Document Control Instance MSR-DCI) has the task of implementing limitations. SwDataDefProps covers various aspects: * Structure of the data element for calibration use cases: is it a single value, a curve, or a map, but also the recordLayouts which specify how such elements are mapped/converted to the DataTypes in the programming language (or in AUTOSAR). This is mainly expressed by properties like swRecordLayout and swCalprmAxisSet * Implementation aspects, mainly expressed by swImplPolicy, swVariableAccessImplPolicy, swAddrMethod, swPointerTagetProps, baseType, implementationDataType and additionalNativeTypeQualifier * Access policy for the MCD system, mainly expressed by swCalibrationAccess * Semantics of the data element, mainly expressed by compuMethod and/or unit, dataConstr, invalidValue * Code generation policy provided by swRecordLayout
M2 Parameter
DataDictionary::DataDefProperties::SwDataDefProps

Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDiagnosisScaling/DcmDspAlternativeDataProps

BSW Parameter	**BSW Type**
DcmDspDataTypeCategory	EcucEnumerationParamDef
BSW Description	
Data category of the alternative Diagnosis Representation.	
Template Description	
The category is a keyword that specializes the semantics of the Identifiable. It affects the expected existence of attributes and the applicability of constraints.	
M2 Parameter	
GenericStructure::GeneralTemplateClasses::Identifiable::Identifiable.category	

Mapping Rule	Mapping Type
The value of the category of the compuMethod referenced by the swDataDefProps of the DiagnosticParameter with the role DiagnosticDataIdentifier.dataElement.	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDiagnosisScaling/DcmDspAlternativeDataProps
BSW Parameter	**BSW Type**
DcmDspLinearScale	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) of an linear scale of the alternative Diagnosis Representation.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDiagnosisScaling/DcmDspAlternativeDataProps/DcmDspLinearScale
BSW Parameter	**BSW Type**
DcmDspDiagnosisRepresentationDataLowerRange	EcucFloatParamDef
BSW Description	
Lower Range for this scale of the data in the alternative Diagnosis Representation.	
Template Description	
This specifies the lower limit of the constraint.	
M2 Parameter	
AsamHdo::Constraints::GlobalConstraints::PhysConstrs.lowerLimit	
Mapping Rule	**Mapping Type**
PhysConstr referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticDataIdentifier.dataElement	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDiagnosisScaling/DcmDspAlternativeDataProps/DcmDspLinearScale
BSW Parameter	**BSW Type**
DcmDspDiagnosisRepresentationDataOffset	EcucFloatParamDef
BSW Description	
Data offset of the alternative Diagnosis Representation for this scale.	
Template Description	
This is the numerator of the rational expression.	

M2 Parameter	
AsamHdo::ComputationMethod::CompuRationalCoeffs.compuNumerator	
Mapping Rule	**Mapping Type**
first compuNumerator of compuMethod referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticDataIdentifier.dataElement	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDiagnosisScaling/DcmDspAlternativeDataProps/DcmDspLinearScale
BSW Parameter	**BSW Type**
DcmDspDiagnosisRepresentationDataResolution	EcucFloatParamDef
BSW Description	
Data resolution of the alternative Diagnosis Representation for this scale.	
Template Description	
This is the numerator of the rational expression.	
M2 Parameter	
AsamHdo::ComputationMethod::CompuRationalCoeffs.compuNumerator	
Mapping Rule	**Mapping Type**
The second compuNumerator in the scope of a CompuMethod referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticDataIdentifier.dataElement defines the scaling factor.	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDiagnosisScaling/DcmDspAlternativeDataProps/DcmDspLinearScale
BSW Parameter	**BSW Type**
DcmDspDiagnosisRepresentationDataUpperRange	EcucFloatParamDef
BSW Description	
Upper Range for this scale of the data in the alternative Diagnosis Representation.	
Template Description	
This specifies the upper limit of the constraint.	
M2 Parameter	
AsamHdo::Constraints::GlobalConstraints::PhysConstrs.upperLimit	
Mapping Rule	**Mapping Type**
PhysConstr referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticDataIdentifier.dataElement	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDiagnosisScaling/DcmDspAlternativeDataProps
BSW Parameter	**BSW Type**
DcmDspTextTableMapping	EcucParamConfContainerDef
BSW Description	

This container contains the configuration (parameters) of the mapping a DataPrototype typed by AutosarDataType that refer to a CompuMethods of category TEXTTABLE or SCALE_LINEAR_AND_TEXTTABLE. Each DcmDspTextTableMapping defines a value pair which is used to map the ECU internal value (DcmDspInternalDataValue) to the vale used in the diagnosis representation (DcmDspDiagnosisRepresentationDataValue) and vice versa. The set of all DcmDspTextTableMappings defines the whole mapping of data.
Template Description
This meta-class represents the ability to express the relationship between a physical value and the mathematical representation. Note that this is still independent of the technical implementation in data types. It only specifies the formula how the internal value corresponds to its physical pendant.
M2 Parameter
AsamHdo::ComputationMethod::CompuMethod

Mapping Rule	**Mapping Type**
This mapping applies if the CompuMethod.category is set to values TEXTTABLE or SCALE_LINEAR_AND_TEXTTABLE.	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDiagnosisScaling/DcmDspAlternativeDataProps/DcmDspTextTableMapping

BSW Parameter	**BSW Type**
DcmDspDiagnosisRepresentationDataValue	EcucIntegerParamDef

BSW Description
The data value in the diagnosis representation.
Template Description
This represents a textual constant in the computation method.
M2 Parameter
AsamHdo::ComputationMethod::CompuConstTextContent.vt

Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDiagnosisScaling/DcmDspAlternativeDataProps/DcmDspTextTableMapping

BSW Parameter	**BSW Type**
DcmDspInternalDataValue	EcucIntegerParamDef

BSW Description
The ECU internal data value.
Template Description
CompuScale.lowerLimit: This specifies the lower limit of the scale. **CompuScale.upperLimit**: This specifies the upper limit of a of the scale.

M2 Parameter	
AsamHdo::ComputationMethod::CompuScale.lowerLimit, AsamHdo::ComputationMethod::CompuScale.upperLimit	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDiagnosisScaling
BSW Parameter	**BSW Type**
DcmDspAlternativeDataType	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) of an alternative Diagnosis Representation by the means of an ApplicationDataType. Additionally the definition of a text table mapping can be a defined for ApplicationDataTypes of category TEXTTABLE and SCALE_LINEAR_AND_TEXTTABLE.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDiagnosisScaling/DcmDspAlternativeDataType
BSW Parameter	**BSW Type**
DcmApplicationDataType	EcucForeignReferenceDef
BSW Description	
Alternative Diagnosis Representation for the data defined by the means of a ApplicationPrimitiveDataType of category VALUE or BOOLEAN.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDiagnosisScaling/DcmDspAlternativeDataType
BSW Parameter	**BSW Type**
DcmDspTextTableMapping	EcucParamConfContainerDef
BSW Description	

This container contains the configuration (parameters) of the mapping a DataPrototype typed by AutosarDataType that refer to a CompuMethods of category TEXTTABLE or SCALE_LINEAR_AND_TEXTTABLE. Each DcmDspTextTableMapping defines a value pair which is used to map the ECU internal value (DcmDspInternalDataValue) to the vale used in the diagnosis representation (DcmDspDiagnosisRepresentationDataValue) and vice versa. The set of all DcmDspTextTableMappings defines the whole mapping of data.
Template Description
This meta-class represents the ability to express the relationship between a physical value and the mathematical representation. Note that this is still independent of the technical implementation in data types. It only specifies the formula how the internal value corresponds to its physical pendant.
M2 Parameter
AsamHdo::ComputationMethod::CompuMethod

Mapping Rule	**Mapping Type**
This mapping applies if the CompuMethod.category is set to values TEXTTABLE or SCALE_LINEAR_AND_TEXTTABLE.	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDiagnosisScaling/DcmDspAlternativeDataType/DcmDspTextTableMapping

BSW Parameter	**BSW Type**
DcmDspDiagnosisRepresentationDataValue	EcucIntegerParamDef

BSW Description
The data value in the diagnosis representation.
Template Description
This represents a textual constant in the computation method.
M2 Parameter
AsamHdo::ComputationMethod::CompuConstTextContent.vt

Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspDiagnosisScaling/DcmDspAlternativeDataType/DcmDspTextTableMapping

BSW Parameter	**BSW Type**
DcmDspInternalDataValue	EcucIntegerParamDef

BSW Description
The ECU internal data value.
Template Description
CompuScale.lowerLimit: This specifies the lower limit of the scale. **CompuScale.upperLimit**: This specifies the upper limit of a of the scale.

M2 Parameter	
AsamHdo::ComputationMethod::CompuScale.lowerLimit, AsamHdo::ComputationMethod::CompuScale.upperLimit	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData
BSW Parameter	**BSW Type**
DcmDspExternalSRDataElementClass	EcucChoiceContainerDef
BSW Description	
This container defines the source of data in a provided port which shall be read respectively the target of data in a required port which shall be written. This container shall contain either one DcmSubElementInDataElementInstance OR DcmDataElementInstance OR DcmSubElementInImplDataElementInstance reference.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspExternalSRDataElement Class
BSW Parameter	**BSW Type**
DcmDataElementInstance	EcucParamConfContainerDef
BSW Description	
Instance Reference to the primitive data in a port where the data element is typed with an ApplicationPrimitveDataType or an ImplementationDataType.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context

Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspExternalSRDataElement Class/DcmDataElementInstance

BSW Parameter	**BSW Type**
DcmDataElementInstanceRef	EcucInstanceReferenceDef
BSW Description	
Instance Reference to the primitive data which shall be read or written. Supported are VariableDataPrototypes in SenderReceiverInterfaces and NvDataInterfaces and ParameterDataPrototypes in ParameterInterfaces (read only). This reference is applicable if the AutosarDataPrototype is typed with a ApplicationPrimitiveDataType of category VALUE or BOOLEAN or if the AutosarDataPrototype is typed with a ImplementationDataType of category VALUE or TYPE_REFERENCE that in turn boils down to VALUE	
Template Description	
This represents the dataElement in the application software that is accessed for diagnostic purpose.	
M2 Parameter	
DiagnosticExtract::ServiceMapping::DiagnosticServiceDataMapping.mappedDataElement	
Mapping Rule	**Mapping Type**
DiagnosticServiceDataMapping maps to a primitive data.	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspExternalSRDataElement Class

BSW Parameter	**BSW Type**
DcmSubElementInDataElementInstance	EcucParamConfContainerDef
BSW Description	
Instance Reference to the primitve sub-element (at any level) of composite data in a port where the data element is typed with an ApplicationCompositeDataType.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspExternalSRDataElement Class/DcmSubElementInDataElementInstance

BSW Parameter	**BSW Type**
DcmSubElementInDataElementInstanceRef	EcucInstanceReferenceDef
BSW Description	
Instance Reference to the primitve sub-element (at any level) of composite data in a port which shall be read or written. Supported are VariableDataPrototypes in SenderReceiverInterfaces and NvDataInterfaces and ParameterDataPrototypes in ParameterInterfaces (read only). This reference is applicable if the AutosarDataPrototype is typed with a ApplicationCompositeDataType.	
Template Description	
This represents the dataElement in the application software that is accessed for diagnostic purpose.	
M2 Parameter	

DiagnosticExtract::ServiceMapping::DiagnosticServiceDataMapping.mappedDataElement	
Mapping Rule	**Mapping Type**
DiagnosticServiceDataMapping maps to a primitive element within a composite data, where the AutosarDataPrototype is typed with a ApplicationComposite DataType.	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspExternalSRDataElement Class
BSW Parameter	**BSW Type**
DcmSubElementInImplDataElementInstance	EcucParamConfContainerDef
BSW Description	
Instance Reference to the primitve sub-element (at any level) of composite data in a port where the data element is typed with an ImplementationDataType.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData/DcmDspExternalSRDataElement Class/DcmSubElementInImplDataElementInstance
BSW Parameter	**BSW Type**
DcmSubElementInImplDataElementInstanceRef	EcucInstanceReferenceDef
BSW Description	
Instance Reference to the primitve sub-element (at any level) of composite data in a port which shall be read or written. Supported are VariableDataPrototypes in SenderReceiverInterfaces and NvDataInterfaces and ParameterDataPrototypes in ParameterInterfaces (read only). This reference is applicable if the AutosarDataPrototype is typed with a ImplementationDataType of category STRUCTURE or ARRAY. Please note that in case of ARRAY the index attribute in the target reference has to be set to select a single array element.	
Template Description	
This represents the dataElement in the application software that is accessed for diagnostic purpose.	
M2 Parameter	
DiagnosticExtract::ServiceMapping::DiagnosticServiceDataMapping.mappedDataElement	
Mapping Rule	**Mapping Type**
DiagnosticServiceDataMapping maps to a primitive element within a composite data, where the AutosarDataPrototype is typed with a ApplicationComposite DataType ImplementationDataType of category STRUCTURE or ARRAY.	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**

Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspData	
BSW Parameter		**BSW Type**
DcmDspOdxDataDescription		EcucAddInfoParamDef
BSW Description		
Defines additional description for ODX documentation		
Template Description		
This specifies the long name of the object. Long name is targeted to human readers and acts like a headline.		
M2 Parameter		
GenericStructure::GeneralTemplateClasses::Identifiable::MultilanguageReferrable.longName		
Mapping Rule		**Mapping Type**
Textual description that characterizes the DID element with respect to the ODX long name can be provided by means of the attribute long-Name.		full
Mapping Status		**Mapping ID**
valid		

BSW Module	**BSW Context**	
Dcm	Dcm/DcmConfigSet/DcmDsp	
BSW Parameter		**BSW Type**
DcmDspDataDefaultEndianness		EcucEnumerationParamDef
BSW Description		
Defines the default endianness belonging to a DID, RID or PID if the corresponding data does not define an endianness.		
Template Description		
Defines the default endianness of the data belonging to a DID or RID which is applicable if the DiagnosticDataElement does not define the endianness via the swDataDefProps.baseType attribute.		
M2 Parameter		
DiagnosticExtract::DiagnosticCommonProps::DiagnosticCommonProps.defaultEndianness		
Mapping Rule		**Mapping Type**
1:1 mapping		full
Mapping Status		**Mapping ID**
valid		

BSW Module	**BSW Context**	
Dcm	Dcm/DcmConfigSet/DcmDsp	
BSW Parameter		**BSW Type**
DcmDspDataInfo		EcucParamConfContainerDef
BSW Description		
This container contains the configuration (parameters) of one Data.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDataInfo

BSW Parameter	BSW Type
DcmDspDataScalingInfoSize	EcucIntegerParamDef
BSW Description	
If Scaling information service is available for this Data, it provides the size in bytes of the scaling information.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp
BSW Parameter	**BSW Type**
DcmDspDid	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) of the DID.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDid
BSW Parameter	**BSW Type**
DcmDspDidIdentifier	EcucIntegerParamDef
BSW Description	
2 byte Identifier of the DID Within each DcmConfigSet all DcmDspDidIdentifier values shall be unique.	
Template Description	
This is the numerical identifier used to identify the DiagnosticAbstractDataIdentifier in the scope of diagnostic workflow	
M2 Parameter	
DiagnosticExtract::CommonDiagnostics::DiagnosticAbstractDataIdentifier.id	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDid
BSW Parameter	**BSW Type**

DcmDspDidInfoRef	EcucReferenceDef
BSW Description	
Reference to DcmDspDidInfo containing information on this DID.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Auto-generation	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDid
BSW Parameter	**BSW Type**
DcmDspDidInfotypeRef	EcucReferenceDef
BSW Description	
Reference to DcmDspVehInfo DcmDspVehInfo contains the configuration (parameters) of the "Request vehicle information service" (service $09).	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDid
BSW Parameter	**BSW Type**
DcmDspDidPidRef	EcucReferenceDef
BSW Description	
Reference to DcmDspPid. DcmDspPid defines the availability of a PID to the DCM	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDid
BSW Parameter	**BSW Type**
DcmDspDidRef	EcucReferenceDef
BSW Description	

Reference to DcmDspDid in case this DID refer to one or serveral other DID's	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDid
BSW Parameter	**BSW Type**
DcmDspDidSignal	EcucParamConfContainerDef
BSW Description	
This container defines the reference to 1 DcmDspData container and position relevant for this DID.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDid/DcmDspDidSignal
BSW Parameter	**BSW Type**
DcmDspDidDataPos	EcucIntegerParamDef
BSW Description	
Defines the position of the data defined by DcmDspDidDataRef reference to DcmDspData container in the DID. The position is defined in bits.	
Template Description	
This represents the bitOffset of the DiagnosticParameter	
M2 Parameter	
DiagnosticExtract::CommonDiagnostics::DiagnosticParameter.bitOffset	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDid/DcmDspDidSignal
BSW Parameter	**BSW Type**
DcmDspDidDataRef	EcucReferenceDef
BSW Description	
Reference to 1 DcmDspData container relevant for this DID.	
Template Description	

M2 Parameter	
Mapping Rule	**Mapping Type**
Auto-generated	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDid
BSW Parameter	**BSW Type**
DcmDspDidUsed	EcucBooleanParamDef
BSW Description	
Allow to activate or deactivate the usage of a DID, for multi purpose ECUs true = DID available false = DID not available	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp
BSW Parameter	**BSW Type**
DcmDspDidInfo	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) of the DID's Info	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDidInfo
BSW Parameter	**BSW Type**
DcmDspDDDIDMaxElements	EcucIntegerParamDef
BSW Description	
Maximum number of source elements of a DDDID.	
Template Description	
This represents the maximum number of source elements of the dynamically created DID.	
M2 Parameter	

DiagnosticExtract::Dcm::DiagnosticService::DynamicallyDefineDataIdentifier::DiagnosticDynamicallyDefineDataIdentifier.maxSourceElement	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDidInfo
BSW Parameter	**BSW Type**
DcmDspDidControl	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) of the DID control.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDidInfo/DcmDspDidControl
BSW Parameter	**BSW Type**
DcmDspDidControlEnableMask	EcucParamConfContainerDef
BSW Description	
The shortname of the container value defines the symbol of the controlMask.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDidInfo/DcmDspDidControl/DcmDspDidControlEnableMask
BSW Parameter	**BSW Type**
DcmDspDidControlMaskBitPosition	EcucIntegerParamDef
BSW Description	
Defines the position of the bit in the controlMask starting from most significant bit (MSB first) to least significant bit. This Bit endianness is identical to the controlMask in UDS. The DcmDspDidControlMaskSize should be considered for most significant bit.	
Template Description	
M2 Parameter	

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDidInfo/DcmDspDidControl
BSW Parameter	**BSW Type**
DcmDspDidControlMask	EcucEnumerationParamDef
BSW Description	
This indicates the presence of "controlEnableMask" in the AUTOSR interface.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDidInfo/DcmDspDidControl
BSW Parameter	**BSW Type**
DcmDspDidControlMaskSize	EcucIntegerParamDef
BSW Description	
The value defines the size of the controlEnableMaskRecord in bytes.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDidInfo/DcmDspDidControl
BSW Parameter	**BSW Type**
DcmDspDidControlModeRuleRef	EcucReferenceDef
BSW Description	
Reference to DcmModeRule Mode rule which controls this DID. If there is no reference, no check of the mode rule shall be done.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**

	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDidInfo/DcmDspDidControl
BSW Parameter	**BSW Type**
DcmDspDidControlSecurityLevelRef	EcucReferenceDef
BSW Description	
Reference to DcmDspSecurityRow Security levels allowed to control this DID. If there is no reference, no check of security level shall be done.	
Template Description	
This represents the associated DiagnosticSecurityLevels	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticAccessPermission.securityLevel	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDidInfo/DcmDspDidControl
BSW Parameter	**BSW Type**
DcmDspDidControlSessionRef	EcucReferenceDef
BSW Description	
Reference to DcmDspSessionRow Sessions allowed to control this DID. If there is no reference, no check of session level shall be done.	
Template Description	
This represents the associated DiagnosticSessions	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticAccessPermission.diagnosticSession	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDidInfo/DcmDspDidControl
BSW Parameter	**BSW Type**
DcmDspDidFreezeCurrentState	EcucBooleanParamDef
BSW Description	
This indicates the presence of "FreezeCurrentState".	
Template Description	
DiagnosticIOControl.freezeCurrentState: Setting this attribute to true represents the ability of the Dcm to execute a freezeCurrentState. **DiagnosticIoControlNeeds.freezeCurrentStateSupported**: This attribute determines, if the referenced port supports temporary freezing of I/O value.	
M2 Parameter	

DiagnosticExtract::Dcm::DiagnosticService::IOControl::DiagnosticIOControl.freezeCurrentState, CommonStructure::ServiceNeeds::DiagnosticIoControlNeeds.freezeCurrentStateSupported	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDidInfo/DcmDspDidControl
BSW Parameter	**BSW Type**
DcmDspDidResetToDefault	EcucBooleanParamDef
BSW Description	
This indicates the presence of "ResetToDefault".	
Template Description	
DiagnosticIOControl.resetToDefault: Setting this attribute to true represents the ability of the Dcm to execute a resetToDefault. **DiagnosticIoControlNeeds.resetToDefaultSupported**: This represents a flag for the existence of the ResetToDefault operation in the service interface.	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticService::IOControl::DiagnosticIOControl.resetToDefault, CommonStructure::ServiceNeeds::DiagnosticIoControlNeeds.resetToDefaultSupported	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDidInfo/DcmDspDidControl
BSW Parameter	**BSW Type**
DcmDspDidShortTermAdjustment	EcucBooleanParamDef
BSW Description	
This indicates the presence of "ShortTermAdjustment".	
Template Description	
DiagnosticIOControl.shortTermAdjustment: Setting this attribute to true represents the ability of the Dcm to execute a shortTermAdjustment. **DiagnosticIoControlNeeds.shortTermAdjustmentSupported**: This attribute determines, if the referenced port supports temporarily setting of I/O value to a specific value provided by the diagnostic tester.	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticService::IOControl::DiagnosticIOControl.shortTermAdjustment, CommonStructure::ServiceNeeds::DiagnosticIoControlNeeds.shortTermAdjustmentSupported	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDidInfo
BSW Parameter	**BSW Type**

DcmDspDidDynamicallyDefined	EcucBooleanParamDef
BSW Description	
Indicates if this DID can be dynamically defined true = DID can be dynamically defined false = DID can not be dynamically defined	
Template Description	
DiagnosticDataIdentifier: This meta-class represents the ability to model a diagnostic data identifier (DID) that is fully specified regarding the payload at configuration-time. **DiagnosticDynamicDataIdentifier**: This meta-class represents the ability to define a diagnostic data identifier (DID) at run-time.	
M2 Parameter	
DiagnosticExtract::CommonDiagnostics::DiagnosticDataIdentifier, DiagnosticExtract::CommonDiagnostics::DiagnosticDynamicDataIdentifier	
Mapping Rule	**Mapping Type**
true: in case the DiagnosticAbstractDataIdentifier for the DID value is aggregated by DiagnosticDynamicDataIdentifier false: in case the DiagnosticAbstract DataIdentifier for the DID value is aggregated by DiagnosticDataIdentifier	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDidInfo
BSW Parameter	**BSW Type**
DcmDspDidRead	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) of the DID read.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDidInfo/DcmDspDidRead
BSW Parameter	**BSW Type**
DcmDspDidReadModeRuleRef	EcucReferenceDef
BSW Description	
Reference to DcmModeRule Mode rule which controls to read this DID. If there is no reference, no check of the mode rule shall be done.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**

	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDidInfo/DcmDspDidRead
BSW Parameter	**BSW Type**
DcmDspDidReadSecurityLevelRef	EcucReferenceDef
BSW Description	
Reference to DcmDspSecurityRow Referenced security levels are allowed to read this DID. If there is no reference, no check of security level shall be done.	
Template Description	
This represents the associated DiagnosticSecurityLevels	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticAccessPermission.securityLevel	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDidInfo/DcmDspDidRead
BSW Parameter	**BSW Type**
DcmDspDidReadSessionRef	EcucReferenceDef
BSW Description	
Reference to DcmDspSessionRow Referenced sessions are allowed to read this DID. If there is no reference, no check of session level shall be done.	
Template Description	
This represents the associated DiagnosticSessions	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticAccessPermission.diagnosticSession	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDidInfo
BSW Parameter	**BSW Type**
DcmDspDidWrite	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) of the DID write.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDidInfo/DcmDspDidWrite
BSW Parameter	**BSW Type**
DcmDspDidWriteModeRuleRef	EcucReferenceDef
BSW Description	
Reference to DcmModeRule Mode rule which controls to write this DID. If there is no reference, no check of the mode rule shall be done.	
Template Description	
M2 Parameter	

Mapping Rule	Mapping Type
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDidInfo/DcmDspDidWrite
BSW Parameter	**BSW Type**
DcmDspDidWriteSecurityLevelRef	EcucReferenceDef
BSW Description	
Reference to DcmDspSecurityRow Referenced security levels are allowed to wirte this DID. If there is no reference, no check of security level shall be done.	
Template Description	
This represents the associated DiagnosticSecurityLevels	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticAccessPermission.securityLevel	

Mapping Rule	Mapping Type
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDidInfo/DcmDspDidWrite
BSW Parameter	**BSW Type**
DcmDspDidWriteSessionRef	EcucReferenceDef
BSW Description	
Reference to DcmDspSessionRow Referenced sessions are allowed to write this DID. If there is no reference, no check of session level shall be done.	
Template Description	
This represents the associated DiagnosticSessions	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticAccessPermission.diagnosticSession	

Mapping Rule	Mapping Type
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp	
BSW Parameter		**BSW Type**
DcmDspDidRange		EcucParamConfContainerDef
BSW Description		
This container defines the DID Range		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDidRange	
BSW Parameter		**BSW Type**
DcmDspDidRangeHasGaps		EcucBooleanParamDef
BSW Description		
Parameter specifying if there are gaps in the DID range (parameter set to TRUE) or not (parameter set to FALSE)		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
		local
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDidRange	
BSW Parameter		**BSW Type**
DcmDspDidRangeIdentifierLowerLimit		EcucIntegerParamDef
BSW Description		
Lower limit of DID range.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
		local
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDidRange	
BSW Parameter		**BSW Type**
DcmDspDidRangeIdentifierUpperLimit		EcucIntegerParamDef
BSW Description		
Upper limit of DID range.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
		local
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDidRange	
BSW Parameter		**BSW Type**
DcmDspDidRangeInfoRef		EcucReferenceDef
BSW Description		
Reference to DcmDspDidInfo containing information on this DID Range.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
		local
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDidRange	
BSW Parameter		**BSW Type**
DcmDspDidRangeIsDidAvailableFnc		EcucFunctionNameDef
BSW Description		
Function name to request from application if a specific DID is available within the range or not. Only relevant if DcmDspDidRangeUsePort is set to false. This parameter is related to the interface Xxx_IsDidAvailable.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
		local
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDidRange	
BSW Parameter		**BSW Type**

DcmDspDidRangeMaxDataLength	EcucIntegerParamDef
BSW Description	
Maximum data length in bytes	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDidRange
BSW Parameter	**BSW Type**
DcmDspDidRangeReadDataLengthFnc	EcucFunctionNameDef
BSW Description	
Function name to request from application the length of the data of a range DID. Only relevant if DcmDspDidRangeUsePort is set to false. This parameter is related to the interface Xxx_ReadDidRangeDataLength.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDidRange
BSW Parameter	**BSW Type**
DcmDspDidRangeReadDidFnc	EcucFunctionNameDef
BSW Description	
Function name to request from application the data range value of a DID.(ReadData-function). Only relevant if DcmDspDidRangeUsePort is set to false. This parameter is related to the interface Xxx_ReadDidData.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDidRange
BSW Parameter	**BSW Type**

DcmDspDidRangeUsePort	EcucBooleanParamDef
BSW Description	
When the parameter DcmDspDidRangeUsePort is set to true the DCM will access the Data using an R-Port requiring a PortInteface DataServices_DIDRange. In that case, DcmDspDidRangeIsDidAvailableFnc, DcmDspDidRangeReadDidFnc and DcmDspDidRangeWriteDidFnc are ignored and the RTE APIs are used. When the parameter DcmDspDidRangeUsePort is false, the DCM calls the functions defined in DcmDspDidRangeIsDidAvailableFnc, DcmDspDidRangeReadDidFnc and DcmDspDidRangeWriteDidFnc.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspDidRange
BSW Parameter	**BSW Type**
DcmDspDidRangeWriteDidFnc	EcucFunctionNameDef
BSW Description	
Function name to request application to write the data range value of a DID.(WriteData-function). Only relevant if DcmDspDidRangeUsePort is set to false. This parameter is related to the interface Xxx_WriteDidData.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp
BSW Parameter	**BSW Type**
DcmDspEnableObdMirror	EcucBooleanParamDef
BSW Description	
DcmDspEnableObdMirror defines whether a DID inside the OBD range (F400-F4FF) and the OBD InfoType range (F800-F8FF) shall get the DID value as defined for OBD on reception of the UDS Service ReadDataByIdentifier (0x22), or not.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp	
BSW Parameter		**BSW Type**
DcmDspMaxDidToRead		EcucIntegerParamDef
BSW Description		
Indicates the maximum allowed DIDs in a single "ReadDataByIdentifier" request.		
Template Description		
This attribute represents the maximum number of allowed DIDs in a single instance of DiagnosticReadDataByIdentifier.		
M2 Parameter		
DiagnosticExtract::Dcm::DiagnosticService::DataByIdentifier::DiagnosticReadDataByIdentifierClass.maxDidToRead		
Mapping Rule		**Mapping Type**
1:1 mapping		full
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp	
BSW Parameter		**BSW Type**
DcmDspMaxPeriodicDidToRead		EcucIntegerParamDef
BSW Description		
Indicates the maximum allowed periodicDIDs which can be read in a single "ReadDataByPeriodicIdentifier" request.		
Template Description		
This represents the maximum number of data identifiers that can be included in one request.		
M2 Parameter		
DiagnosticExtract::Dcm::DiagnosticService::ReadDataByPeriodicID::DiagnosticReadDataByPeriodicIDClass.maxPeriodicDidToRead		
Mapping Rule		**Mapping Type**
1:1 mapping		full
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp	
BSW Parameter		**BSW Type**
DcmDspMemory		EcucParamConfContainerDef
BSW Description		
This container contains the configuration of the memory access.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context

Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspMemory	
BSW Parameter		**BSW Type**
DcmDspAddressAndLengthFormatIdentifier		EcucParamConfContainerDef
BSW Description		
This container contains the configuration of the supported AddressAndLengthFormatIdentifiers for memory access.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	**BSW Context**	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspMemory/DcmDspAddressAndLength FormatIdentifier	
BSW Parameter		**BSW Type**
DcmDspSupportedAddressAndLengthFormatIdentifier		EcucIntegerParamDef
BSW Description		
This parameter defines the supported AddressAndLengthFormatIdentifier of the request message.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
		local
Mapping Status		**Mapping ID**
valid		

BSW Module	**BSW Context**	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspMemory	
BSW Parameter		**BSW Type**
DcmDspMemoryIdInfo		EcucParamConfContainerDef
BSW Description		
Provides the value of memory identifier used to select the desired memory device This container contains the configuration of the memory access requested through diagnostic services : ReadMemoryByAddress, WriteMemoryByAddress, RequestDownload, RequestUpload		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	**BSW Context**

Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspMemory/DcmDspMemoryIdInfo	
BSW Parameter		**BSW Type**
DcmDspMemoryIdValue		EcucIntegerParamDef
BSW Description		
Value of the memory device identifier used. If this parameter is not configured, the DCM will not use MemoryIdentifier parameter. The Dcm_WriteMemory and Dcm_ReadMemory callouts shall be called without the MemoryIdentifier parameter. If this parameter is configured, the DCM will use MemoryIdentifier parameter to select the memory device to use. The Dcm_WriteMemory and Dcm_ReadMemory callouts shall be called with the MemoryIdentifier parameter. Every values configured in the configuration parameter DcmDspMemoryIdValue shall be unique. The MemoryIdValue is retrieved from the request messages (RMBA,WMBA,RD,RU,DDDI) according to ISO-14229-1.		
Template Description		
This represents the identification of the memory segment.		
M2 Parameter		
DiagnosticExtract::Dcm::DiagnosticService::MemoryByAddress::DiagnosticMemoryIdentifier.id		
Mapping Rule		**Mapping Type**
1:1 mapping		full
Mapping Status		**Mapping ID**
valid		

BSW Module	**BSW Context**	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspMemory/DcmDspMemoryIdInfo	
BSW Parameter		**BSW Type**
DcmDspReadMemoryRangeInfo		EcucParamConfContainerDef
BSW Description		
Provides the range of memory address allowed for reading		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	**BSW Context**	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspMemory/DcmDspMemoryIdInfo/DcmDspReadMemoryRangeInfo	
BSW Parameter		**BSW Type**
DcmDspReadMemoryRangeHigh		EcucIntegerParamDef
BSW Description		
High memory address of a range allowed for reading		
Template Description		
This represents the upper bound for addresses of the memory segment.		
M2 Parameter		

DiagnosticExtract::Dcm::DiagnosticService::MemoryByAddress::DiagnosticMemoryIdentifier.memoryHighAddress	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspMemory/DcmDspMemoryIdInfo/DcmDspReadMemoryRangeInfo
BSW Parameter	**BSW Type**
DcmDspReadMemoryRangeLow	EcucIntegerParamDef
BSW Description	
Low memory address of a range allowed for reading	
Template Description	
This represents the lower bound for addresses of the memory segment.	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticService::MemoryByAddress::DiagnosticMemoryIdentifier.memoryLowAddress	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspMemory/DcmDspMemoryIdInfo/DcmDspReadMemoryRangeInfo
BSW Parameter	**BSW Type**
DcmDspReadMemoryRangeModeRuleRef	EcucReferenceDef
BSW Description	
Reference to DcmModeRule Mode rule which controls read access on this memory address. If there is no reference, no check of the mode rule shall be done.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspMemory/DcmDspMemoryIdInfo/DcmDspReadMemoryRangeInfo
BSW Parameter	**BSW Type**
DcmDspReadMemoryRangeSecurityLevelRef	EcucReferenceDef
BSW Description	
Link to the Security Access Levels needed for read access on this memory address. If there is no reference, no check of security level shall be done.	

Template Description	
This represents the associated DiagnosticSecurityLevels	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticAccessPermission.securityLevel	
Mapping Rule	**Mapping Type**
DiagnosticMemoryIdentifier referenced in the role memoryRange is referenced by a DiagnosticReadMemoryByAddress The accessPermission holds the security level information.	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspMemory/DcmDspMemoryIdInfo
BSW Parameter	**BSW Type**
DcmDspWriteMemoryRangeInfo	EcucParamConfContainerDef
BSW Description	
Provides the range of memory address allowed for writting.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspMemory/DcmDspMemoryIdInfo/DcmDspWriteMemoryRangeInfo
BSW Parameter	**BSW Type**
DcmDspWriteMemoryRangeHigh	EcucIntegerParamDef
BSW Description	
High memory address of a range allowed for writting.	
Template Description	
This represents the upper bound for addresses of the memory segment.	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticService::MemoryByAddress::DiagnosticMemoryIdentifier.memoryHighAddress	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspMemory/DcmDspMemoryIdInfo/DcmDspWriteMemoryRangeInfo
BSW Parameter	**BSW Type**
DcmDspWriteMemoryRangeLow	EcucIntegerParamDef
BSW Description	
Low memory address of a range allowed for writting	

Template Description	
This represents the lower bound for addresses of the memory segment.	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticService::MemoryByAddress::DiagnosticMemoryIdentifier.memoryLowAddress	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspMemory/DcmDspMemoryIdInfo/DcmDspWriteMemoryRangeInfo
BSW Parameter	**BSW Type**
DcmDspWriteMemoryRangeModeRuleRef	EcucReferenceDef
BSW Description	
Reference to DcmModeRule Mode rule which controls write access on this memory address. If there is no reference, no check of the mode rule shall be done.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspMemory/DcmDspMemoryIdInfo/DcmDspWriteMemoryRangeInfo
BSW Parameter	**BSW Type**
DcmDspWriteMemoryRangeSecurityLevelRef	EcucReferenceDef
BSW Description	
Link to the Security Access Levels needed for write access on this memory address. If there is no reference, no check of security level shall be done.	
Template Description	
This represents the associated DiagnosticSecurityLevels	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticAccessPermission.securityLevel	
Mapping Rule	**Mapping Type**
DiagnosticMemoryIdentifier referenced in the role memoryRange is referenced by a DiagnosticWriteMemoryByAddress The accessPermission holds the security level information.	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp
BSW Parameter	**BSW Type**

DcmDspPeriodicDidTransmission	EcucParamConfContainerDef
BSW Description	
This container contains the configuration for the Periodic Did transmission. This container exists only if the UDS Service ReadDataByPeriodicIdentifier(0x2A) is configured.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspPeriodicDidTransmission
BSW Parameter	**BSW Type**
DcmDspMaxPeriodicDidScheduler	EcucIntegerParamDef
BSW Description	
Defines the maximum number of periodicDataIdentifiers that can be scheduled concurrently.	
Template Description	
This represents the maximum number of periodic data identifiers that can be scheduled in parallel.	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticService::ReadDataByPeriodicID::DiagnosticReadDataByPeriodicIDClass.schedulerMaxNumber	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp
BSW Parameter	**BSW Type**
DcmDspPeriodicTransmission	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) for Periodic Transmission Scheduler.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspPeriodicTransmission
BSW Parameter	**BSW Type**
DcmDspPeriodicTransmissionFastRate	EcucFloatParamDef
BSW Description	

This parameter give the transmission rate of the requested periodicDataIdentifiers to be used if the parameter transmissionMode given in the ReadDataByPeriodicID request is equal to 0x03 ("sendAtFastRate"). This parameter value in seconds have to be configured as a multiple of DcmTaskTime. min: A negative value and zero is not allowed.
Template Description
This represents the period of the DiagnosticPeriodicRate in seconds.
M2 Parameter
DiagnosticExtract::Dcm::DiagnosticService::ReadDataByPeriodicID::DiagnosticPeriodicRate.period

Mapping Rule	**Mapping Type**
In case DiagnosticPeriodicRate.category is set to PERIODIC_RATE_FAST	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspPeriodicTransmission

BSW Parameter	**BSW Type**
DcmDspPeriodicTransmissionMediumRate	EcucFloatParamDef

BSW Description
This parameter give the transmission rate of the requested periodicDataIdentifiers to be used if the parameter transmissionMode given in the ReadDataByPeriodicID request is equal to 0x02 ("sendAtMediumRate"). This parameter value in seconds have to be configured as a multiple of DcmTaskTime. min: A negative value and zero is not allowed.
Template Description
This represents the period of the DiagnosticPeriodicRate in seconds.
M2 Parameter
DiagnosticExtract::Dcm::DiagnosticService::ReadDataByPeriodicID::DiagnosticPeriodicRate.period

Mapping Rule	**Mapping Type**
In case DiagnosticPeriodicRate.category is set to PERIODIC_RATE_MEDIUM	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspPeriodicTransmission

BSW Parameter	**BSW Type**
DcmDspPeriodicTransmissionSlowRate	EcucFloatParamDef

BSW Description
This parameter give the transmission rate of the requested periodicDataIdentifiers to be used if the parameter transmissionMode given in the ReadDataByPeriodicID request is equal to 0x01 ("sendAtSlowRate"). This parameter value in seconds have to be configured as a multiple of DcmTaskTime. min: A negative value and zero is not allowed.
Template Description
This represents the period of the DiagnosticPeriodicRate in seconds.
M2 Parameter

DiagnosticExtract::Dcm::DiagnosticService::ReadDataByPeriodicID::DiagnosticPeriodicRate.period	
Mapping Rule	**Mapping Type**
In case DiagnosticPeriodicRate.category is set to PERIODIC_RATE_SLOW	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp
BSW Parameter	**BSW Type**
DcmDspPid	EcucParamConfContainerDef
BSW Description	
This container defines the availability of a PID to the DCM.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspPid
BSW Parameter	**BSW Type**
DcmDspPidData	EcucParamConfContainerDef
BSW Description	
This container defines the parameter for a Signal in the PID.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspPid/DcmDspPidData
BSW Parameter	**BSW Type**
DcmDspPidDataPos	EcucIntegerParamDef
BSW Description	
This is the position in bit of the PID structure and will not start at position 0 in case a support information is available (for packeted PIDs).	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**

valid	

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspPid/DcmDspPidData	
BSW Parameter		**BSW Type**
DcmDspPidDataSize		EcucIntegerParamDef
BSW Description		
Length of data associated to the PID in bit(s).		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspPid/DcmDspPidData	
BSW Parameter		**BSW Type**
DcmDspPidDataSupportInfo		EcucParamConfContainerDef
BSW Description		
This container defines the supported information.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspPid/DcmDspPidData/DcmDspPidData SupportInfo	
BSW Parameter		**BSW Type**
DcmDspPidDataSupportInfoBit		EcucIntegerParamDef
BSW Description		
Referenced Bit of the SupportInfo		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context

Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspPid/DcmDspPidData/DcmDspPidDataSupportInfo

BSW Parameter	**BSW Type**
DcmDspPidDataSupportInfoRef	EcucReferenceDef
BSW Description	
Reference to DcmDspPidSupportInfo	
Template Description	
M2 Parameter	

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspPid/DcmDspPidData

BSW Parameter	**BSW Type**
DcmDspPidService01	EcucParamConfContainerDef
BSW Description	
Contains specific configuration parameter of PID for service $01. This container exists only if DcmDspPidService is set to DCM_SERVICE_01 or DCM_SERVICE_01_02.	
Template Description	
M2 Parameter	

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspPid/DcmDspPidData/DcmDspPidService01

BSW Parameter	**BSW Type**
DcmDspPidDataEndianness	EcucEnumerationParamDef
BSW Description	
Defines the endianness of the data belonging to a PID in a diagnostic response message. If no DcmDspPidDataEndianness is defined the value of DcmDspDataDefaultEndianness is applicable.	
Template Description	
M2 Parameter	

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspPid/DcmDspPidData/DcmDspPidService01

BSW Parameter	BSW Type
DcmDspPidDataReadFnc	EcucFunctionNameDef

BSW Description

Function name for reading PID data value.
This is only relevant if DcmDspPidDataUsePort==USE_DATA_SYNCH_FNC.

This parameter is related to the interface Xxx_ReadData.

Template Description

M2 Parameter

Mapping Rule	Mapping Type
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspPid/DcmDspPidData/DcmDspPidService01

BSW Parameter	BSW Type
DcmDspPidDataType	EcucEnumerationParamDef

BSW Description

Provide the implementation data type of data belonging to a PID.

Template Description

M2 Parameter

Mapping Rule	Mapping Type
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspPid/DcmDspPidData/DcmDspPidService01/DcmDspPidDataType

BSW Parameter	BSW Type
SINT16_N	EcucEnumerationLiteralDef

BSW Description

Type of the data is sint16 array.

Template Description

BaseTypeDirectDefinition.baseTypeSize:
Describes the length of the data type specified in the container in bits.

BaseTypeDirectDefinition.baseTypeEncoding:
This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence.

M2 Parameter

AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize,
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding

Mapping Rule	Mapping Type
baseTypeEncoding = 2C baseTypeSize = 16	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspPid/DcmDspPidData/DcmDspPidService01/DcmDspPidDataType
BSW Parameter	**BSW Type**
SINT32_N	EcucEnumerationLiteralDef
BSW Description	
Type of the data is sint32 array.	
Template Description	
BaseTypeDirectDefinition.baseTypeSize: Describes the length of the data type specified in the container in bits. **BaseTypeDirectDefinition.baseTypeEncoding**: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding	
Mapping Rule	**Mapping Type**
baseTypeEncoding = 2C baseTypeSize = 32	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspPid/DcmDspPidData/DcmDspPidService01/DcmDspPidDataType
BSW Parameter	**BSW Type**
SINT8_N	EcucEnumerationLiteralDef
BSW Description	
Type of the data is sint8 array.	
Template Description	
BaseTypeDirectDefinition.baseTypeSize: Describes the length of the data type specified in the container in bits. **BaseTypeDirectDefinition.baseTypeEncoding**: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding	
Mapping Rule	**Mapping Type**
baseTypeEncoding = 2C baseTypeSize = 8	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspPid/DcmDspPidData/DcmDspPidService01/DcmDspPidDataType

BSW Parameter	BSW Type
UINT16_N	EcucEnumerationLiteralDef

BSW Description

Type of the data is uint16 array.

Template Description

BaseTypeDirectDefinition.baseTypeEncoding:
This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence.

BaseTypeDirectDefinition.baseTypeSize:
Describes the length of the data type specified in the container in bits.

M2 Parameter

AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding,
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize

Mapping Rule	Mapping Type
baseTypeEncoding = NONE baseTypeSize = 16	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspPid/DcmDspPidData/DcmDspPidService01/DcmDspPidDataType

BSW Parameter	BSW Type
UINT32_N	EcucEnumerationLiteralDef

BSW Description

Type of the data is uint32 array.

Template Description

BaseTypeDirectDefinition.baseTypeEncoding:
This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence.

BaseTypeDirectDefinition.baseTypeSize:
Describes the length of the data type specified in the container in bits.

M2 Parameter

AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding,
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize

Mapping Rule	Mapping Type
baseTypeEncoding = NONE baseTypeSize = 32	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspPid/DcmDspPidData/DcmDspPidService01/DcmDspPidDataType

BSW Parameter	BSW Type
UINT8_DYN	EcucEnumerationLiteralDef

BSW Description

Type of the data is uint8 array with dynamic length.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
baseTypeEncoding = NONE baseTypeSize = 8	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspPid/DcmDspPidData/DcmDspPidService01/DcmDspPidDataType
BSW Parameter	**BSW Type**
UINT8_N	EcucEnumerationLiteralDef
BSW Description	
Type of the data is uint8 array.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
baseTypeEncoding = NONE baseTypeSize = 8	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspPid/DcmDspPidData/DcmDspPidService01
BSW Parameter	**BSW Type**
DcmDspPidDataUsePort	EcucEnumerationParamDef
BSW Description	

If this parameter is set to USE_DATA_SYNCH_FNC, the DCM will use the function defined in DcmDspPidDataReadFnc to get the PID data value. If this parameter is set to USE_DATA_SYNCH_CLIENT_SERVER, the DCM will have an R-Port requiring the interface DataServices_{Data}. If this parameter is set to USE_DATA_SENDER_RECEIVER, the DCM will have an R-Port requiring a SenderReceiverInterface The R-Port is named DataServices_{Data} where {Data} is the name of the container DcmDspPidData.
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspPid/DcmDspPidData

BSW Parameter	**BSW Type**
DcmDspPidService02	EcucParamConfContainerDef

BSW Description
Contains specific configuration parameter of PID for service $02. This container exists only if DcmDspPidService is set to DCM_SERVICE_02 or DCM_SERVICE_01_02.
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspPid/DcmDspPidData/DcmDspPidService02

BSW Parameter	**BSW Type**
DcmDspPidDataDemRef	EcucReferenceDef

BSW Description
Reference to DemPidDataElement in DEM configuration. Allows to link the DCM PID and DEM PID configuration for Mode $02.
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspPid
BSW Parameter	**BSW Type**
DcmDspPidIdentifier	EcucIntegerParamDef
BSW Description	
1 byte Identifier of the PID Within each DcmConfigSet all DcmDspPidIdentifier values shall be unique.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspPid
BSW Parameter	**BSW Type**
DcmDspPidService	EcucEnumerationParamDef
BSW Description	
Indicates if a PID is used with service $01 and/or $02	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspPid
BSW Parameter	**BSW Type**
DcmDspPidSize	EcucIntegerParamDef
BSW Description	
Length of a PID in byte(s).	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspPid
BSW Parameter	**BSW Type**

DcmDspPidSupportInfo	EcucParamConfContainerDef
BSW Description	
This container defines the support information (typically byte A) to declare the usability of the data bytes within the so-called packeted PIDs (e.g. PID$68).	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspPid/DcmDspPidSupportInfo
BSW Parameter	**BSW Type**
DcmDspPidSupportInfoLen	EcucIntegerParamDef
BSW Description	
Length of the support information in bytes.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspPid/DcmDspPidSupportInfo
BSW Parameter	**BSW Type**
DcmDspPidSupportInfoPos	EcucIntegerParamDef
BSW Description	
Position of the support information in bytes.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspPid
BSW Parameter	**BSW Type**
DcmDspPidUsed	EcucBooleanParamDef
BSW Description	

Allow to activate or deactivate the usage of a PID, for multi purpose ECUs true = PID is available false = PID is not available
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp

BSW Parameter	**BSW Type**
DcmDspPowerDownTime	EcucIntegerParamDef

BSW Description
This parameter indicates to the client the minimum time of the stand-by sequence the server will remain in the power-down sequence. The resolution of this parameter is one second per count. The following values are valid: 00 - FE hex: 0 - 254 s powerDownTime; FF hex: indicates a failure or time not available. This value needs to be defined by the integrator according to the ECU capabilities. This parameter has to be available if the service EcuReset, sub-service enableRapidPowerShutDown is configured.
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp

BSW Parameter	**BSW Type**
DcmDspRequestControl	EcucParamConfContainerDef

BSW Description
This container contains the configuration (parameters) of the "Request control of on-board system, test or component" service (Service $08). The DCM will request the control using an R-Port requiring a PortInteface RequestControlServices_{Tid}. The R-Port is named RequestControlServices_{Tid} where {Tid} is the name of the container DcmDspRequestControl.
Template Description

M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRequestControl
BSW Parameter	**BSW Type**
DcmDspRequestControlInBufferSize	EcucIntegerParamDef
BSW Description	
Number of bytes to be provided in the input buffer of the interface RequestControlServices_{Tid} for OBD Service $08	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRequestControl
BSW Parameter	**BSW Type**
DcmDspRequestControlOutBufferSize	EcucIntegerParamDef
BSW Description	
Number of bytes to be provided in the output buffer of the interface RequestControlServices_{Tid} for OBD Service $08	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRequestControl
BSW Parameter	**BSW Type**
DcmDspRequestControlTestId	EcucIntegerParamDef
BSW Description	
Test Id for Service $08	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**

Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp
BSW Parameter	**BSW Type**
DcmDspRequestFileTransfer	EcucParamConfContainerDef
BSW Description	
This container contains the configuration for RequestFileTransfer. This container only exists if RequestFileTransfer is configured.	
Template Description	
This diagnostic service instance implements the UDS service 0x38.	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticService::RequestFileTransfer::DiagnosticRequestFileTransfer	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRequestFileTransfer
BSW Parameter	**BSW Type**
DcmRequestFileTransferFileSizeParameterLength	EcucIntegerParamDef
BSW Description	
Length of the fileSizeCompressed and fileSizeUncompressedOrDirInfoLength in the Dcm_ProcessRequestFileTransfer operation and response message.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRequestFileTransfer
BSW Parameter	**BSW Type**
DcmRequestFileTransferLengthFormatIdentifier	EcucIntegerParamDef
BSW Description	
Defines the length (number of bytes) of the maxNumberOfBlockLength parameter.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp	
BSW Parameter		**BSW Type**
DcmDspRoe		EcucParamConfContainerDef
BSW Description		
Provide the configuration of the ResponseOnEvent mechanism.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoe	
BSW Parameter		**BSW Type**
DcmDspRoeEvent		EcucParamConfContainerDef
BSW Description		
This container contains a list of all supported Roe eventTypeRecords which are accepted by this ECU. At most one DcmDspRoeEvent container is allowed to define a DcmDspRoeEventProperties container with the choice DcmDspRoeOnDTCStatusChange.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoe/DcmDspRoeEvent	
BSW Parameter		**BSW Type**
DcmDspRoeEventId		EcucIntegerParamDef
BSW Description		
EventId for a global identification of this ROE event it is used within APIs Dcm_TriggerOnEvent() and the ModeDeclarationGroup. The ratio Ids should be sequentially ordered beginning with 0 and no gaps in between.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Auto-generation		local
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoe/DcmDspRoeEvent
BSW Parameter	**BSW Type**
DcmDspRoeEventProperties	EcucChoiceContainerDef
BSW Description	
This container contains the properties of Roe eventTypeRecords. In one DcmDspRoeEventProperties container one DcmDspRoeOnDTCStatusChange or DcmDspRoeOnChangeOfDataIdentifier container shall be defined.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoe/DcmDspRoeEvent/DcmDspRoeEventProperties
BSW Parameter	**BSW Type**
DcmDspRoeOnChangeOfDataIdentifier	EcucParamConfContainerDef
BSW Description	
This container contains configuration of a eventTypeRecord onChangeOfDataIdentifier accepted by this ECU.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoe/DcmDspRoeEvent/DcmDspRoeEventProperties/DcmDspRoeOnChangeOfDataIdentifier
BSW Parameter	**BSW Type**
DcmDspRoeDidRef	EcucReferenceDef
BSW Description	
Reference to a Did which is watched.	
Template Description	
This represents the corresponding DiagnosticDataIdentifier.	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticService::ResponseOnEvent::DiagnosticDataChangeTrigger.dataIdentifier	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoe/DcmDspRoeEvent/DcmDspRoe EventProperties
BSW Parameter	**BSW Type**
DcmDspRoeOnDTCStatusChange	EcucParamConfContainerDef
BSW Description	
This container contains configuration of a eventTypeRecord onDTCStatusChange accepted by this ECU. Please note that currently are no additional parameters for DcmDspRoeOnDTCStatusChange are defined. Therefore the existence of the container denotes the choice.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoe/DcmDspRoeEvent
BSW Parameter	**BSW Type**
DcmDspRoeInitialEventStatus	EcucEnumerationParamDef
BSW Description	
Initial Roe status of this RoeEvent	
Template Description	
This represents the initial status of the enclosing DiagnosticResponseOnEventTrigger.	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticService::ResponseOnEvent::DiagnosticResponseOnEvent Trigger.initialEventStatus	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoe/DcmDspRoeEvent/DcmDspRoe InitialEventStatus
BSW Parameter	**BSW Type**
DCM_ROE_CLEARED	EcucEnumerationLiteralDef
BSW Description	
Template Description	
This means that the ResponseOnEvent is initially cleared.	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticService::ResponseOnEvent::DiagnosticInitialEventStatus Enum.returnOnEventCleared	
Mapping Rule	**Mapping Type**
1:1 mapping	full

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoe/DcmDspRoeEvent/DcmDspRoe InitialEventStatus	
BSW Parameter		**BSW Type**
DCM_ROE_STOPPED		EcucEnumerationLiteralDef
BSW Description		
Template Description		
This means that the ResponseOnEvent is initially stopped.		
M2 Parameter		
DiagnosticExtract::Dcm::DiagnosticService::ResponseOnEvent::DiagnosticInitialEventStatus Enum.returnOnEventStopped		
Mapping Rule		**Mapping Type**
1:1 mapping		full
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoe	
BSW Parameter		**BSW Type**
DcmDspRoeEventWindowTime		EcucParamConfContainerDef
BSW Description		
This container configures the available EventWindowTime in this Ecu. This container contains a sub-set of EventWindowTimes supported by the Dcm, to limit the Ecu resources.		
Template Description		
This attribute clarifies the validity of the eventWindow		
M2 Parameter		
DiagnosticExtract::Dcm::DiagnosticService::ResponseOnEvent::DiagnosticEventWindow.event WindowTime		
Mapping Rule		**Mapping Type**
1:1 mapping		full
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoe/DcmDspRoeEventWindowTime	
BSW Parameter		**BSW Type**
DcmDspRoeEventWindowTime		EcucEnumerationParamDef
BSW Description		
Value of the EventWindowTime		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**

Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoe/DcmDspRoeEventWindowTime/ DcmDspRoeEventWindowTime
BSW Parameter	**BSW Type**
DCM_ROE_EVENT_WINDOW_CURRENT_AND_FOLLOWING_CYCLE	EcucEnumerationLiteralDef
BSW Description	
Template Description	
This means that the window extends to this and the following cycle.	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticService::ResponseOnEvent::DiagnosticEventWindowTimeEnum.eventWindowCurrentAndFollowingCycle	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoe/DcmDspRoeEventWindowTime/ DcmDspRoeEventWindowTime
BSW Parameter	**BSW Type**
DCM_ROE_EVENT_WINDOW_CURRENT_CYCLE	EcucEnumerationLiteralDef
BSW Description	
Template Description	
This means that the window is limited to the current cycle.	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticService::ResponseOnEvent::DiagnosticEventWindowTimeEnum.eventWindowCurrentCycle	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoe/DcmDspRoeEventWindowTime/ DcmDspRoeEventWindowTime
BSW Parameter	**BSW Type**
DCM_ROE_EVENT_WINDOW_INFINITE	EcucEnumerationLiteralDef
BSW Description	
Template Description	
This means that the window extents without a border.	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticService::ResponseOnEvent::DiagnosticEventWindowTimeEnum.eventWindowInfinite	

Mapping Rule	Mapping Type
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoe/DcmDspRoeEventWindowTime
BSW Parameter	**BSW Type**
DcmDspRoeStorageState	EcucBooleanParamDef
BSW Description	
If this parameter is set to TRUE the StorageStateBit will be evaluated if this EventWindowTime is requested.	
Template Description	
If this attribute is set to TRUE the StorageStateBit will be evaluated if this EventWindowTime is requested.	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticService::ResponseOnEvent::DiagnosticEventWindow.storage StateEvaluation	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoe
BSW Parameter	**BSW Type**
DcmDspRoeInterMessageTime	EcucFloatParamDef
BSW Description	
Provide the minimum time in seconds between two transmissions of ROE event. It is used for the delay between two different consecutive Roe transmissions.	
Template Description	
Provide the minimum time in seconds between two consecutive transmissions of an ROE event.	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticService::ResponseOnEvent::DiagnosticResponseOnEvent Class.interMessageTime	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp
BSW Parameter	**BSW Type**
DcmDspRoutine	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) for Routines	
Template Description	
M2 Parameter	

Mapping Rule	Mapping Type
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine
BSW Parameter	**BSW Type**
DcmDspCommonAuthorizationRef	EcucReferenceDef
BSW Description	
Reference to DcmDspCommonAuthorization Common authorization configuration taken from the referenced DcmDspCommonAuthorization. If there is no reference, no check on the commonly defined authorization conditions shall be done.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine
BSW Parameter	**BSW Type**
DcmDspRequestRoutineResults	EcucParamConfContainerDef
BSW Description	
Provides the configuration of RequestResult subservice for RoutineControl service. Existence indicates that the RequestRoutineResults in the RoutineControl is supported.	
Template Description	
This represents the ability to request the result of a running routine.	
M2 Parameter	
DiagnosticExtract::CommonDiagnostics::DiagnosticRoutine.requestResult	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults
BSW Parameter	**BSW Type**
DcmDspRequestRoutineResultsCommonAuthorizationRef	EcucReferenceDef
BSW Description	

Reference to DcmDspCommonAuthorization Common authorization configuration taken from the referenced DcmDspRequestRoutineResultsCommonAuthorizationRef. If there is no reference, no check on the commonly defined authorization conditions shall be done to get the routine result.
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults

BSW Parameter	**BSW Type**
DcmDspRequestRoutineResultsFnc	EcucFunctionNameDef

BSW Description
Function name for request to application the results of a routine. (Routine_RequestResults-function) This parameter is related to the interface Xxx_RequestResults.
Template Description
Specialization of ServiceDependency in the context of an BswInternalBehavior. It allows to associate BswModuleEntries and data defined for a BSW module or cluster to a given ServiceNeeds element.
M2 Parameter
BswModuleTemplate::BswBehavior::BswServiceDependency

Mapping Rule	**Mapping Type**
It could be possible to get the FNC name via BswServiceDependency	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults

BSW Parameter	**BSW Type**
DcmDspRequestRoutineResultsOut	EcucParamConfContainerDef

BSW Description
Provide description of output parameter of RequestResult subservice for RoutineControl service.
Template Description
This represents the response parameters.
M2 Parameter
DiagnosticExtract::CommonDiagnostics::DiagnosticRequestRoutineResults.response

Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**

Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults/DcmDspRequestRoutineResultsOut

BSW Parameter	BSW Type
DcmDspRequestRoutineResultsOutSignal	EcucParamConfContainerDef

BSW Description

Provides description of a routine signal used in RoutineControl service.

The ordering defined via the index attribute of the subcontainers in this list represents the order of the dataOutN elements in the XXX_RequestResult function call.

Template Description

This represents the related dataElement of the DiagnosticParameter

M2 Parameter

DiagnosticExtract::CommonDiagnostics::DiagnosticParameter.dataElement

Mapping Rule	Mapping Type
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults/DcmDspRequestRoutineResultsOut/DcmDspRequestRoutineResultsOutSignal

BSW Parameter	BSW Type
DcmDspArgumentScaling	EcucChoiceContainerDef

BSW Description

This container contains the configuration (arguments) of an alternative Diagnosis Representation. Out if this the scaling between Diagnosis and ECU internal representation and vice versa can be calculated.

Template Description

M2 Parameter

Mapping Rule	Mapping Type
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults/DcmDspRequestRoutineResultsOut/DcmDspRequestRoutineResultsOutSignal/DcmDspArgumentScaling

BSW Parameter	BSW Type
DcmDspAlternativeArgumentData	EcucParamConfContainerDef

BSW Description

This container contains the configuration (parameters) of an alternative Diagnosis Representation by the means of a ArgumentDataPrototype.

Template Description

M2 Parameter

Mapping Rule	Mapping Type

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults/DcmDspRequestRoutineResultsOut/DcmDspRequestRoutineResultsOutSignal/DcmDspArgumentScaling/DcmDspAlternativeArgumentData

BSW Parameter	BSW Type
DcmDataElement	EcucForeignReferenceDef

BSW Description
Alternative Diagnosis Representation for the data defined by the means of a ArgumentDataPrototype

Template Description

M2 Parameter

Mapping Rule	Mapping Type
	local

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults/DcmDspRequestRoutineResultsOut/DcmDspRequestRoutineResultsOutSignal/DcmDspArgumentScaling

BSW Parameter	BSW Type
DcmDspAlternativeDataProps	EcucParamConfContainerDef

BSW Description
This container contains the configuration (parameters) of an alternative Diagnosis Representation by the means of ECU configuration parameters. The physical unit of the alternative data representation is defined by the DataPrototype referenced by DcmDspExternalSRDataElementClass. Additionally the definition of a text table mapping can be a defined for DcmDspDataTypeCategory TEXTTABLE and SCALE_LINEAR_AND_TEXTTABLE

Template Description

This class is a collection of properties relevant for data objects under various aspects. One could consider this class as a "pattern of inheritance by aggregation". The properties can be applied to all objects of all classes in which SwDataDefProps is aggregated. Note that not all of the attributes or associated elements are useful all of the time. Hence, the process definition (e.g. expressed with an OCL or a Document Control Instance MSR-DCI) has the task of implementing limitations. SwDataDefProps covers various aspects: * Structure of the data element for calibration use cases: is it a single value, a curve, or a map, but also the recordLayouts which specify how such elements are mapped/converted to the DataTypes in the programming language (or in AUTOSAR). This is mainly expressed by properties like swRecordLayout and swCalprmAxisSet * Implementation aspects, mainly expressed by swImplPolicy, swVariableAccessImplPolicy, swAddrMethod, swPointerTagetProps, baseType, implementationDataType and additionalNativeTypeQualifier * Access policy for the MCD system, mainly expressed by swCalibrationAccess * Semantics of the data element, mainly expressed by compuMethod and/or unit, dataConstr, invalidValue * Code generation policy provided by swRecordLayout
M2 Parameter
DataDictionary::DataDefProperties::SwDataDefProps

Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults/DcmDspRequestRoutineResultsOut/DcmDspRequestRoutineResultsOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps

BSW Parameter	**BSW Type**
DcmDspDataTypeCategory	EcucEnumerationParamDef
BSW Description	
Data category of the alternative Diagnosis Representation.	
Template Description	
The category is a keyword that specializes the semantics of the Identifiable. It affects the expected existence of attributes and the applicability of constraints.	
M2 Parameter	
GenericStructure::GeneralTemplateClasses::Identifiable::Identifiable.category	

Mapping Rule	**Mapping Type**
The value of the category of the compuMethod referenced by the swDataDefProps of the DiagnosticParameter with the role DiagnosticDataIdentifier.dataElement.	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults/DcmDspRequestRoutineResultsOut/DcmDspRequestRoutineResultsOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps

BSW Parameter	BSW Type
DcmDspLinearScale	EcucParamConfContainerDef

BSW Description
This container contains the configuration (parameters) of an linear scale of the alternative Diagnosis Representation.

Template Description

M2 Parameter

Mapping Rule	Mapping Type

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults/DcmDspRequestRoutineResultsOut/DcmDspRequestRoutineResultsOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps/DcmDspLinearScale

BSW Parameter	BSW Type
DcmDspDiagnosisRepresentationDataLowerRange	EcucFloatParamDef

BSW Description
Lower Range for this scale of the data in the alternative Diagnosis Representation.

Template Description
This specifies the lower limit of the constraint.

M2 Parameter
AsamHdo::Constraints::GlobalConstraints::PhysConstrs.lowerLimit

Mapping Rule	Mapping Type
PhysConstr referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticDataIdentifier.dataElement	full

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults/DcmDspRequestRoutineResultsOut/DcmDspRequestRoutineResultsOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps/DcmDspLinearScale

BSW Parameter	BSW Type
DcmDspDiagnosisRepresentationDataOffset	EcucFloatParamDef

BSW Description
Data offset of the alternative Diagnosis Representation for this scale.

Template Description
This is the numerator of the rational expression.

M2 Parameter
AsamHdo::ComputationMethod::CompuRationalCoeffs.compuNumerator

Mapping Rule	Mapping Type

first compuNumerator of compuMethod referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticDataIdentifier.dataElement	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults/DcmDspRequestRoutineResultsOut/DcmDspRequestRoutineResultsOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps/DcmDspLinearScale

BSW Parameter	**BSW Type**
DcmDspDiagnosisRepresentationDataResolution	EcucFloatParamDef
BSW Description	
Data resolution of the alternative Diagnosis Representation for this scale.	
Template Description	
This is the numerator of the rational expression.	
M2 Parameter	
AsamHdo::ComputationMethod::CompuRationalCoeffs.compuNumerator	
Mapping Rule	**Mapping Type**
The second compuNumerator in the scope of a CompuMethod referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticDataIdentifier.dataElement defines the scaling factor.	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults/DcmDspRequestRoutineResultsOut/DcmDspRequestRoutineResultsOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps/DcmDspLinearScale

BSW Parameter	**BSW Type**
DcmDspDiagnosisRepresentationDataUpperRange	EcucFloatParamDef
BSW Description	
Upper Range for this scale of the data in the alternative Diagnosis Representation.	
Template Description	
This specifies the upper limit of the constraint.	
M2 Parameter	
AsamHdo::Constraints::GlobalConstraints::PhysConstrs.upperLimit	
Mapping Rule	**Mapping Type**
PhysConstr referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticDataIdentifier.dataElement	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults/DcmDspRequestRoutineResultsOut/DcmDspRequestRoutineResultsOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps

BSW Parameter	**BSW Type**
DcmDspTextTableMapping	EcucParamConfContainerDef
BSW Description	

This container contains the configuration (parameters) of the mapping a DataPrototype typed by AutosarDataType that refer to a CompuMethods of category TEXTTABLE or SCALE_LINEAR_AND_TEXTTABLE. Each DcmDspTextTableMapping defines a value pair which is used to map the ECU internal value (DcmDspInternalDataValue) to the vale used in the diagnosis representation (DcmDspDiagnosisRepresentationDataValue) and vice versa. The set of all DcmDspTextTableMappings defines the whole mapping of data.
Template Description
This meta-class represents the ability to express the relationship between a physical value and the mathematical representation. Note that this is still independent of the technical implementation in data types. It only specifies the formula how the internal value corresponds to its physical pendant.
M2 Parameter
AsamHdo::ComputationMethod::CompuMethod

Mapping Rule	**Mapping Type**
This mapping applies if the CompuMethod.category is set to values TEXTTABLE or SCALE_LINEAR_AND_TEXTTABLE.	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults/DcmDspRequestRoutineResultsOut/DcmDspRequestRoutineResultsOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps/DcmDspTextTableMapping

BSW Parameter	**BSW Type**
DcmDspDiagnosisRepresentationDataValue	EcucIntegerParamDef
BSW Description	
The data value in the diagnosis representation.	
Template Description	
This represents a textual constant in the computation method.	
M2 Parameter	
AsamHdo::ComputationMethod::CompuConstTextContent.vt	

Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults/DcmDspRequestRoutineResultsOut/DcmDspRequestRoutineResultsOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps/DcmDspTextTableMapping

BSW Parameter	**BSW Type**
DcmDspInternalDataValue	EcucIntegerParamDef
BSW Description	
The ECU internal data value.	
Template Description	

CompuScale.lowerLimit: This specifies the lower limit of the scale. **CompuScale.upperLimit**: This specifies the upper limit of a of the scale.
M2 Parameter
AsamHdo::ComputationMethod::CompuScale.lowerLimit, AsamHdo::ComputationMethod::CompuScale.upperLimit

Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults/DcmDspRequestRoutineResultsOut/DcmDspRequestRoutineResultsOutSignal/DcmDspArgumentScaling

BSW Parameter	**BSW Type**
DcmDspAlternativeDataType	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) of an alternative Diagnosis Representation by the means of an ApplicationDataType. Additionally the definition of a text table mapping can be a defined for ApplicationDataTypes of category TEXTTABLE and SCALE_LINEAR_AND_TEXTTABLE.	
Template Description	
M2 Parameter	

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults/DcmDspRequestRoutineResultsOut/DcmDspRequestRoutineResultsOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataType

BSW Parameter	**BSW Type**
DcmApplicationDataType	EcucForeignReferenceDef
BSW Description	
Alternative Diagnosis Representation for the data defined by the means of a ApplicationPrimitiveDataType of category VALUE or BOOLEAN.	
Template Description	
M2 Parameter	

Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults/DcmDspRequestRoutineResultsOut/DcmDspRequestRoutineResultsOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataType

BSW Parameter	BSW Type
DcmDspTextTableMapping	EcucParamConfContainerDef

BSW Description

This container contains the configuration (parameters) of the mapping a DataPrototype typed by AutosarDataType that refer to a CompuMethods of category TEXTTABLE or SCALE_LINEAR_AND_TEXTTABLE.

Each DcmDspTextTableMapping defines a value pair which is used to map the ECU internal value (DcmDspInternalDataValue) to the vale used in the diagnosis representation (DcmDspDiagnosisRepresentationDataValue) and vice versa.

The set of all DcmDspTextTableMappings defines the whole mapping of data.

Template Description

This meta-class represents the ability to express the relationship between a physical value and the mathematical representation.

Note that this is still independent of the technical implementation in data types. It only specifies the formula how the internal value corresponds to its physical pendant.

M2 Parameter

AsamHdo::ComputationMethod::CompuMethod

Mapping Rule	Mapping Type
This mapping applies if the CompuMethod.category is set to values TEXTTABLE or SCALE_LINEAR_AND_TEXTTABLE.	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults/DcmDspRequestRoutineResultsOut/DcmDspRequestRoutineResultsOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataType/DcmDspTextTableMapping

BSW Parameter	BSW Type
DcmDspDiagnosisRepresentationDataValue	EcucIntegerParamDef

BSW Description

The data value in the diagnosis representation.

Template Description

This represents a textual constant in the computation method.

M2 Parameter

AsamHdo::ComputationMethod::CompuConstTextContent.vt

Mapping Rule	Mapping Type
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context

Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults/DcmDspRequestRoutineResultsOut/DcmDspRequestRoutineResultsOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataType/DcmDspTextTableMapping

BSW Parameter	**BSW Type**
DcmDspInternalDataValue	EcucIntegerParamDef
BSW Description	
The ECU internal data value.	
Template Description	
CompuScale.lowerLimit: This specifies the lower limit of the scale. **CompuScale.upperLimit**: This specifies the upper limit of a of the scale.	
M2 Parameter	
AsamHdo::ComputationMethod::CompuScale.lowerLimit, AsamHdo::ComputationMethod::CompuScale.upperLimit	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults/DcmDspRequestRoutineResultsOut/DcmDspRequestRoutineResultsOutSignal

BSW Parameter	**BSW Type**
DcmDspRoutineSignalEndianness	EcucEnumerationParamDef
BSW Description	
Defines the endianness of the data belonging to a Routine Out Signal for RequestResult subfunction. If no DcmDspRoutineSignalEndianness is defined the value of DcmDspDataDefaultEndianness is applicable.	
Template Description	
This attribute specifies the byte order of the base type.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.byteOrder	
Mapping Rule	**Mapping Type**
baseType.baseTypeDefinition.byteOrder referenced by swDataDefProps of the DiagnosticParameter with the role DiagnostictRequestRoutineResult.response	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults/DcmDspRequestRoutineResultsOut/DcmDspRequestRoutineResultsOutSignal

BSW Parameter	**BSW Type**
DcmDspRoutineSignalLength	EcucIntegerParamDef
BSW Description	
Provide the length in bits of the signal in the RoutineControl request/response	
Template Description	

The existence of this attribute turns the data instance into an array of data. The attribute determines the size of the array in terms of how many elements the array can take.	
M2 Parameter	
DiagnosticExtract::CommonDiagnostics::DiagnosticDataElement.maxNumberOfElements	
Mapping Rule	**Mapping Type**
Only in case of variable length required (according to constr_6008). Calculation: DcmDspRoutineSignalLength = maxNumberOfElements * 8	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults/DcmDspRequestRoutineResultsOut/DcmDspRequestRoutineResultsOutSignal
BSW Parameter	**BSW Type**
DcmDspRoutineSignalPos	EcucIntegerParamDef
BSW Description	
Provide the position of the signal in the RoutineControl request/response. The position is defined in bits.	
Template Description	
This represents the bitOffset of the DiagnosticParameter	
M2 Parameter	
DiagnosticExtract::CommonDiagnostics::DiagnosticParameter.bitOffset	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults/DcmDspRequestRoutineResultsOut/DcmDspRequestRoutineResultsOutSignal
BSW Parameter	**BSW Type**
DcmDspRoutineSignalType	EcucEnumerationParamDef
BSW Description	
Provide the type of the signal in the RoutineControl request/response.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults/DcmDspRequestRoutineResultsOut/DcmDspRequestRoutineResultsOutSignal/DcmDspRoutineSignalType
BSW Parameter	**BSW Type**
BOOLEAN	EcucEnumerationLiteralDef

BSW Description	
Type of the signal is boolean.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticRequestRoutineResults.response baseTypeEncoding = BOOLEAN baseTypeSize = 1	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults/DcmDspRequestRoutineResultsOut/DcmDspRequestRoutineResultsOutSignal/DcmDspRoutineSignalType

BSW Parameter	BSW Type
SINT16	EcucEnumerationLiteralDef

BSW Description	
Type of the signal is sint16.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticRequestRoutineResults.response baseTypeEncoding = 2C baseTypeSize = 16	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults/DcmDspRequestRoutineResultsOut/DcmDspRequestRoutineResultsOutSignal/DcmDspRoutineSignalType

BSW Parameter	BSW Type
SINT32	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is sint32.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticRequestRoutineResults.response baseTypeEncoding = 2C baseTypeSize = 32	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults/DcmDspRequestRoutineResultsOut/DcmDspRequestRoutineResultsOutSignal/DcmDspRoutineSignalType
BSW Parameter	**BSW Type**
SINT8	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is sint8.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticRequestRoutineResults.response baseTypeEncoding = 2C baseTypeSize = 8	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context

Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults/DcmDspRequestRoutineResultsOut/DcmDspRequestRoutineResultsOutSignal/DcmDspRoutineSignalType

BSW Parameter	**BSW Type**
UINT16	EcucEnumerationLiteralDef

BSW Description

Type of the signal is uint16.

Template Description

BaseTypeDirectDefinition.baseTypeEncoding:
This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence.

BaseTypeDirectDefinition.baseTypeSize:
Describes the length of the data type specified in the container in bits.

M2 Parameter

AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding,
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize

Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticRequestRoutineResults.response baseTypeEncoding = NONE baseTypeSize = 16	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults/DcmDspRequestRoutineResultsOut/DcmDspRequestRoutineResultsOutSignal/DcmDspRoutineSignalType

BSW Parameter	**BSW Type**
UINT32	EcucEnumerationLiteralDef

BSW Description

Type of the signal is uint32.

Template Description

BaseTypeDirectDefinition.baseTypeEncoding:
This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence.

BaseTypeDirectDefinition.baseTypeSize:
Describes the length of the data type specified in the container in bits.

M2 Parameter

AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding,
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize

Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticRequestRoutineResults.response baseTypeEncoding = NONE baseTypeSize = 32	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults/DcmDspRequestRoutineResultsOut/DcmDspRequestRoutineResultsOutSignal/DcmDspRoutineSignalType

BSW Parameter	BSW Type
UINT8	EcucEnumerationLiteralDef

BSW Description

Type of the signal is uint8.

Template Description

BaseTypeDirectDefinition.baseTypeEncoding:
This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence.

BaseTypeDirectDefinition.baseTypeSize:
Describes the length of the data type specified in the container in bits.

M2 Parameter

AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding,
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize

Mapping Rule	Mapping Type
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticRequestRoutineResults.response baseTypeEncoding = NONE baseTypeSize = 8	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspRequestRoutineResults/DcmDspRequestRoutineResultsOut/DcmDspRequestRoutineResultsOutSignal/DcmDspRoutineSignalType

BSW Parameter	BSW Type
VARIABLE_LENGTH	EcucEnumerationLiteralDef

BSW Description

Type of the signal is uint8[(DcmDspRoutineSignalLength+7)/8].

This is only valid for the last signal and when DcmDspRoutineSignalType is set to VARIABLE_LENGTH.

Template Description

BaseTypeDirectDefinition.baseTypeEncoding:
This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence.

BaseTypeDirectDefinition.baseTypeSize:
Describes the length of the data type specified in the container in bits.

M2 Parameter

AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding,
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize

Mapping Rule	Mapping Type

referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticRequestRoutineResults.response baseTypeEncoding = NONE baseTypeSize = 8 arraySizeSemantics = variableSize	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine
BSW Parameter	**BSW Type**
DcmDspRoutineIdentifier	EcucIntegerParamDef
BSW Description	
2 bytes Identifier of the RID Within each DcmConfigSet all DcmDspRoutineIdentifier values shall be unique.	
Template Description	
This is the numerical identifier used to identify the DiagnosticRoutine in the scope of diagnostic workflow	
M2 Parameter	
DiagnosticExtract::CommonDiagnostics::DiagnosticRoutine.id	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine
BSW Parameter	**BSW Type**
DcmDspRoutineInfoByte	EcucIntegerParamDef
BSW Description	
Manufacturer specific value reported to the tester for the record identifiers 0xE000 to OxE1FF. (OBD use cases)	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine
BSW Parameter	**BSW Type**
DcmDspRoutineTidRef	EcucReferenceDef
BSW Description	
Reference to DcmDspRequestControl DcmDspRequestControl contains the configuration (parameters) of the "Request control of on-board system, test or component" service (Service $08).	

Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine
BSW Parameter	**BSW Type**
DcmDspRoutineUsePort	EcucBooleanParamDef
BSW Description	
If this parameter is set to true, the DCM uses a port requiring a PortInterface RoutineServices_{RoutineName}. The R-Port is named RoutineServices_{RoutineName} where {RoutineName} is the name of the container DcmDspRoutine In that case, the configuration must not provide function names in DcmDspStartRoutineFnc, DcmDspStopRoutineFnc or DcmDspRequestResultsRoutineFnc. If this is false, the DCM expects to find the names of the functions to be used in DcmDspStartRoutineFnc, DcmDspStopRoutineFnc or DcmDspRequestResultsRoutineFnc.	
Template Description	
This represents the ability to define a mapping of a diagnostic service to a software-component. This kind of service mapping is applicable for the usage of ClientServerInterfaces.	
M2 Parameter	
DiagnosticExtract::ServiceMapping::DiagnosticServiceSwMapping	
Mapping Rule	**Mapping Type**
TRUE: DiagnosticServiceSwMapping is having a SwcServiceDependency FALSE: DiagnosticServiceSwMapping is having a BswServiceDependency	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine
BSW Parameter	**BSW Type**
DcmDspRoutineUsed	EcucBooleanParamDef
BSW Description	
Allow to activate or deactivate the usage of a Routine, for multi purpose ECUs True = Routine is available False = Routine is not available	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine	
BSW Parameter		**BSW Type**
DcmDspStartRoutine		EcucParamConfContainerDef
BSW Description		
Provides the configuration of Start subservice for RoutineControl service.		
Template Description		
This represents the ability to start a routine		
M2 Parameter		
DiagnosticExtract::CommonDiagnostics::DiagnosticRoutine.start		
Mapping Rule		**Mapping Type**
1:1 mapping		full
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine	
BSW Parameter		**BSW Type**
DcmDspStartRoutineCommonAuthorizationRef		EcucReferenceDef
BSW Description		
Reference to DcmDspCommonAuthorization Common authorization configuration taken from the referenced DcmDspStartRoutineCommonAuthorizationRef. If there is no reference, no check on the commonly defined authorization conditions shall be done to start the routine.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine	
BSW Parameter		**BSW Type**
DcmDspStartRoutineFnc		EcucFunctionNameDef
BSW Description		
Function name for request to application to start a routine. (Routine_Start-function) This parameter is related to the interface Xxx_Start.		
Template Description		
This is supposed to represent a reference to a BswServiceDependency. the latter is not derived from Referrable and therefore this detour needs to be implemented to still let BswServiceDependency become the target of a reference.		
M2 Parameter		
DiagnosticExtract::ServiceMapping::DiagnosticServiceSwMapping.mappedBswServiceDependency		
Mapping Rule		**Mapping Type**
1:1 mapping		full

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine
BSW Parameter	**BSW Type**
DcmDspStartRoutineIn	EcucParamConfContainerDef
BSW Description	
Provide description of input parameter of Start subservice for RoutineControl service	
Template Description	
This represents the request parameters.	
M2 Parameter	
DiagnosticExtract::CommonDiagnostics::DiagnosticStartRoutine.request	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDsp StartRoutineIn
BSW Parameter	**BSW Type**
DcmDspStartRoutineInSignal	EcucParamConfContainerDef
BSW Description	
Provide description of a routine signal used in RoutineControl service. The ordering defined via the index attribute of the subcontainers in this list represents the order of the dataInN elements in the XXX_Start function call.	
Template Description	
This represents the related dataElement of the DiagnosticParameter	
M2 Parameter	
DiagnosticExtract::CommonDiagnostics::DiagnosticParameter.dataElement	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDsp StartRoutineIn/DcmDspStartRoutineInSignal
BSW Parameter	**BSW Type**
DcmDspArgumentScaling	EcucChoiceContainerDef
BSW Description	
This container contains the configuration (arguments) of an alternative Diagnosis Representation. Out if this the scaling between Diagnosis and ECU internal representation and vice versa can be calculated.	
Template Description	
M2 Parameter	

Mapping Rule	Mapping Type
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineIn/DcmDspStartRoutineInSignal/DcmDspArgumentScaling
BSW Parameter	**BSW Type**
DcmDspAlternativeArgumentData	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) of an alternative Diagnosis Representation by the means of a ArgumentDataPrototype.	
Template Description	
M2 Parameter	

Mapping Rule	Mapping Type
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineIn/DcmDspStartRoutineInSignal/DcmDspArgumentScaling/DcmDspAlternativeArgumentData
BSW Parameter	**BSW Type**
DcmDataElement	EcucForeignReferenceDef
BSW Description	
Alternative Diagnosis Representation for the data defined by the means of a ArgumentDataPrototype	
Template Description	
M2 Parameter	

Mapping Rule	Mapping Type
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineIn/DcmDspStartRoutineInSignal/DcmDspArgumentScaling
BSW Parameter	**BSW Type**
DcmDspAlternativeDataProps	EcucParamConfContainerDef
BSW Description	

This container contains the configuration (parameters) of an alternative Diagnosis Representation by the means of ECU configuration parameters. The physical unit of the alternative data representation is defined by the DataPrototype referenced by DcmDspExternalSRDataElementClass. Additionally the definition of a text table mapping can be a defined for DcmDspDataTypeCategory TEXTTABLE and SCALE_LINEAR_AND_TEXTTABLE
Template Description
This class is a collection of properties relevant for data objects under various aspects. One could consider this class as a "pattern of inheritance by aggregation". The properties can be applied to all objects of all classes in which SwDataDefProps is aggregated. Note that not all of the attributes or associated elements are useful all of the time. Hence, the process definition (e.g. expressed with an OCL or a Document Control Instance MSR-DCI) has the task of implementing limitations. SwDataDefProps covers various aspects: * Structure of the data element for calibration use cases: is it a single value, a curve, or a map, but also the recordLayouts which specify how such elements are mapped/converted to the DataTypes in the programming language (or in AUTOSAR). This is mainly expressed by properties like swRecordLayout and swCalprmAxisSet * Implementation aspects, mainly expressed by swImplPolicy, swVariableAccessImplPolicy, swAddrMethod, swPointerTagetProps, baseType, implementationDataType and additionalNativeTypeQualifier * Access policy for the MCD system, mainly expressed by swCalibrationAccess * Semantics of the data element, mainly expressed by compuMethod and/or unit, dataConstr, invalidValue * Code generation policy provided by swRecordLayout
M2 Parameter
DataDictionary::DataDefProperties::SwDataDefProps

Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineIn/DcmDspStartRoutineInSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps

BSW Parameter	**BSW Type**
DcmDspDataTypeCategory	EcucEnumerationParamDef
BSW Description	
Data category of the alternative Diagnosis Representation.	
Template Description	
The category is a keyword that specializes the semantics of the Identifiable. It affects the expected existence of attributes and the applicability of constraints.	
M2 Parameter	

GenericStructure::GeneralTemplateClasses::Identifiable::Identifiable.category	
Mapping Rule	**Mapping Type**
The value of the category of the compuMethod referenced by the swDataDefProps of the DiagnosticParameter with the role DiagnosticDataIdentifier.dataElement.	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineIn/DcmDspStartRoutineInSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps
BSW Parameter	**BSW Type**
DcmDspLinearScale	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) of an linear scale of the alternative Diagnosis Representation.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineIn/DcmDspStartRoutineInSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps/DcmDspLinearScale
BSW Parameter	**BSW Type**
DcmDspDiagnosisRepresentationDataLowerRange	EcucFloatParamDef
BSW Description	
Lower Range for this scale of the data in the alternative Diagnosis Representation.	
Template Description	
This specifies the lower limit of the constraint.	
M2 Parameter	
AsamHdo::Constraints::GlobalConstraints::PhysConstrs.lowerLimit	
Mapping Rule	**Mapping Type**
PhysConstr referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticDataIdentifier.dataElement	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineIn/DcmDspStartRoutineInSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps/DcmDspLinearScale
BSW Parameter	**BSW Type**
DcmDspDiagnosisRepresentationDataOffset	EcucFloatParamDef
BSW Description	

Data offset of the alternative Diagnosis Representation for this scale.	
Template Description	
This is the numerator of the rational expression.	
M2 Parameter	
AsamHdo::ComputationMethod::CompuRationalCoeffs.compuNumerator	
Mapping Rule	**Mapping Type**
first compuNumerator of compuMethod referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticDataIdentifier.dataElement	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineIn/DcmDspStartRoutineInSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps/DcmDspLinearScale
BSW Parameter	**BSW Type**
DcmDspDiagnosisRepresentationDataResolution	EcucFloatParamDef
BSW Description	
Data resolution of the alternative Diagnosis Representation for this scale.	
Template Description	
This is the numerator of the rational expression.	
M2 Parameter	
AsamHdo::ComputationMethod::CompuRationalCoeffs.compuNumerator	
Mapping Rule	**Mapping Type**
The second compuNumerator in the scope of a CompuMethod referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticDataIdentifier.dataElement defines the scaling factor.	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineIn/DcmDspStartRoutineInSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps/DcmDspLinearScale
BSW Parameter	**BSW Type**
DcmDspDiagnosisRepresentationDataUpperRange	EcucFloatParamDef
BSW Description	
Upper Range for this scale of the data in the alternative Diagnosis Representation.	
Template Description	
This specifies the upper limit of the constraint.	
M2 Parameter	
AsamHdo::Constraints::GlobalConstraints::PhysConstrs.upperLimit	
Mapping Rule	**Mapping Type**
PhysConstr referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticDataIdentifier.dataElement	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**

Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineIn/DcmDspStartRoutineInSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps

BSW Parameter	BSW Type
DcmDspTextTableMapping	EcucParamConfContainerDef

BSW Description

This container contains the configuration (parameters) of the mapping a DataPrototype typed by AutosarDataType that refer to a CompuMethods of category TEXTTABLE or SCALE_LINEAR_AND_TEXTTABLE.

Each DcmDspTextTableMapping defines a value pair which is used to map the ECU internal value (DcmDspInternalDataValue) to the vale used in the diagnosis representation (DcmDspDiagnosisRepresentationDataValue) and vice versa.

The set of all DcmDspTextTableMappings defines the whole mapping of data.

Template Description

This meta-class represents the ability to express the relationship between a physical value and the mathematical representation.

Note that this is still independent of the technical implementation in data types. It only specifies the formula how the internal value corresponds to its physical pendant.

M2 Parameter

AsamHdo::ComputationMethod::CompuMethod

Mapping Rule	Mapping Type
This mapping applies if the CompuMethod.category is set to values TEXTTABLE or SCALE_LINEAR_AND_TEXTTABLE.	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineIn/DcmDspStartRoutineInSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps/DcmDspTextTableMapping

BSW Parameter	BSW Type
DcmDspDiagnosisRepresentationDataValue	EcucIntegerParamDef

BSW Description

The data value in the diagnosis representation.

Template Description

This represents a textual constant in the computation method.

M2 Parameter

AsamHdo::ComputationMethod::CompuConstTextContent.vt

Mapping Rule	Mapping Type
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineIn/DcmDspStartRoutineInSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps/DcmDspTextTableMapping

BSW Parameter	BSW Type
DcmDspInternalDataValue	EcucIntegerParamDef

BSW Description	
The ECU internal data value.	
Template Description	
CompuScale.lowerLimit: This specifies the lower limit of the scale. **CompuScale.upperLimit**: This specifies the upper limit of a of the scale.	
M2 Parameter	
AsamHdo::ComputationMethod::CompuScale.lowerLimit, AsamHdo::ComputationMethod::CompuScale.upperLimit	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineIn/DcmDspStartRoutineInSignal/DcmDspArgumentScaling
BSW Parameter	**BSW Type**
DcmDspAlternativeDataType	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) of an alternative Diagnosis Representation by the means of an ApplicationDataType. Additionally the definition of a text table mapping can be a defined for ApplicationDataTypes of category TEXTTABLE and SCALE_LINEAR_AND_TEXTTABLE.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineIn/DcmDspStartRoutineInSignal/DcmDspArgumentScaling/DcmDspAlternativeDataType
BSW Parameter	**BSW Type**
DcmApplicationDataType	EcucForeignReferenceDef
BSW Description	
Alternative Diagnosis Representation for the data defined by the means of a ApplicationPrimitiveDataType of category VALUE or BOOLEAN.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**

valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineIn/DcmDspStartRoutineInSignal/DcmDspArgumentScaling/DcmDspAlternativeDataType

BSW Parameter	BSW Type
DcmDspTextTableMapping	EcucParamConfContainerDef

BSW Description

This container contains the configuration (parameters) of the mapping a DataPrototype typed by AutosarDataType that refer to a CompuMethods of category TEXTTABLE or SCALE_LINEAR_AND_TEXTTABLE.

Each DcmDspTextTableMapping defines a value pair which is used to map the ECU internal value (DcmDspInternalDataValue) to the vale used in the diagnosis representation (DcmDspDiagnosisRepresentationDataValue) and vice versa.

The set of all DcmDspTextTableMappings defines the whole mapping of data.

Template Description

This meta-class represents the ability to express the relationship between a physical value and the mathematical representation.

Note that this is still independent of the technical implementation in data types. It only specifies the formula how the internal value corresponds to its physical pendant.

M2 Parameter

AsamHdo::ComputationMethod::CompuMethod

Mapping Rule	Mapping Type
This mapping applies if the CompuMethod.category is set to values TEXTTABLE or SCALE_LINEAR_AND_TEXTTABLE.	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineIn/DcmDspStartRoutineInSignal/DcmDspArgumentScaling/DcmDspAlternativeDataType/DcmDspTextTableMapping

BSW Parameter	BSW Type
DcmDspDiagnosisRepresentationDataValue	EcucIntegerParamDef

BSW Description

The data value in the diagnosis representation.

Template Description

This represents a textual constant in the computation method.

M2 Parameter

AsamHdo::ComputationMethod::CompuConstTextContent.vt

Mapping Rule	Mapping Type
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context

Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineIn/DcmDspStartRoutineInSignal/DcmDspArgumentScaling/DcmDspAlternativeDataType/DcmDspTextTableMapping

BSW Parameter	**BSW Type**
DcmDspInternalDataValue	EcucIntegerParamDef
BSW Description	
The ECU internal data value.	
Template Description	
CompuScale.lowerLimit: This specifies the lower limit of the scale. **CompuScale.upperLimit**: This specifies the upper limit of a of the scale.	
M2 Parameter	
AsamHdo::ComputationMethod::CompuScale.lowerLimit, AsamHdo::ComputationMethod::CompuScale.upperLimit	

Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineIn/DcmDspStartRoutineInSignal

BSW Parameter	**BSW Type**
DcmDspRoutineSignalEndianness	EcucEnumerationParamDef
BSW Description	
Defines the endianness of the data belonging to a Routine In Signal for Start subfunction. If no DcmDspRoutineSignalEndianness is defined the value of DcmDspDataDefaultEndianness is applicable.	
Template Description	
This attribute specifies the byte order of the base type.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.byteOrder	

Mapping Rule	**Mapping Type**
baseType.baseTypeDefinition.byteOrder referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStartRoutine.request.	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineIn/DcmDspStartRoutineInSignal

BSW Parameter	**BSW Type**
DcmDspRoutineSignalLength	EcucIntegerParamDef
BSW Description	
Provide the length in bits of the signal in the RoutineControl request/response	
Template Description	
The existence of this attribute turns the data instance into an array of data. The attribute determines the size of the array in terms of how many elements the array can take.	
M2 Parameter	

DiagnosticExtract::CommonDiagnostics::DiagnosticDataElement.maxNumberOfElements	
Mapping Rule	**Mapping Type**
"Only in case of variable length required (according to constr_6008). Calculation: DcmDspRoutineSignalLength = maxNumberOfElements * 8.	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineIn/DcmDspStartRoutineInSignal
BSW Parameter	**BSW Type**
DcmDspRoutineSignalPos	EcucIntegerParamDef
BSW Description	
Provide the position of the signal in the RoutineControl request/response. The position is defined in bits.	
Template Description	
This represents the bitOffset of the DiagnosticParameter	
M2 Parameter	
DiagnosticExtract::CommonDiagnostics::DiagnosticParameter.bitOffset	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineIn/DcmDspStartRoutineInSignal
BSW Parameter	**BSW Type**
DcmDspRoutineSignalType	EcucEnumerationParamDef
BSW Description	
Provide the type of the signal in the RoutineControl request/response.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineIn/DcmDspStartRoutineInSignal/DcmDspRoutineSignalType
BSW Parameter	**BSW Type**
BOOLEAN	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is boolean.	
Template Description	

BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.
M2 Parameter
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize

Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStartRoutine.request baseTypeEncoding = BOOLEAN baseTypeSize = 1	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineIn/DcmDspStartRoutineInSignal/DcmDspRoutineSignalType

BSW Parameter	**BSW Type**
SINT16	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is sint16.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	

Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStartRoutine.request baseTypeEncoding = 2C baseTypeSize = 16	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineIn/DcmDspStartRoutineInSignal/DcmDspRoutineSignalType

BSW Parameter	**BSW Type**
SINT32	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is sint32.	
Template Description	

BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStartRoutine.request baseTypeEncoding = 2C baseTypeSize = 32	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineIn/DcmDspStartRoutineInSignal/DcmDspRoutineSignalType
BSW Parameter	**BSW Type**
SINT8	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is sint8.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStartRoutine.request baseTypeEncoding = 2C baseTypeSize = 8	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineIn/DcmDspStartRoutineInSignal/DcmDspRoutineSignalType
BSW Parameter	**BSW Type**
UINT16	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is uint16.	
Template Description	

BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStartRoutine.request baseTypeEncoding = NONE baseTypeSize = 16	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineIn/DcmDspStartRoutineInSignal/DcmDspRoutineSignalType
BSW Parameter	**BSW Type**
UINT32	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is uint32.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStartRoutine.request baseTypeEncoding = NONE baseTypeSize = 32	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineIn/DcmDspStartRoutineInSignal/DcmDspRoutineSignalType
BSW Parameter	**BSW Type**
UINT8	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is uint8.	
Template Description	

BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.
M2 Parameter
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize

Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStartRoutine.request baseTypeEncoding = NONE baseTypeSize = 8	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineIn/DcmDspStartRoutineInSignal/DcmDspRoutineSignalType

BSW Parameter	**BSW Type**
VARIABLE_LENGTH	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is uint8[(DcmDspRoutineSignalLength+7)/8]. This is only valid for the last signal and when DcmDspRoutineSignalType is set to VARIABLE_LENGTH.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	

Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStartRoutine.request baseTypeEncoding = NONE baseTypeSize = 8 arraySizeSemantics = variableSize	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine

BSW Parameter	**BSW Type**
DcmDspStartRoutineOut	EcucParamConfContainerDef

BSW Description	
Provide description of output parameter of Start subservice for RoutineControl service.	
Template Description	
This represents the response parameters.	
M2 Parameter	
DiagnosticExtract::CommonDiagnostics::DiagnosticStartRoutine.response	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineOut
BSW Parameter	**BSW Type**
DcmDspStartRoutineOutSignal	EcucParamConfContainerDef
BSW Description	
Provide description of a routine signal used in RoutineControl service. The ordering defined via the index attribute of the subcontainers in this list represents the order of the dataOutN elements in the XXX_Start function call.	
Template Description	
This represents the related dataElement of the DiagnosticParameter	
M2 Parameter	
DiagnosticExtract::CommonDiagnostics::DiagnosticParameter.dataElement	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineOut/DcmDspStartRoutineOutSignal
BSW Parameter	**BSW Type**
DcmDspArgumentScaling	EcucChoiceContainerDef
BSW Description	
This container contains the configuration (arguments) of an alternative Diagnosis Representation. Out if this the scaling between Diagnosis and ECU internal representation and vice versa can be calculated.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineOut/DcmDspStartRoutineOutSignal/DcmDspArgumentScaling

BSW Parameter	BSW Type
DcmDspAlternativeArgumentData	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) of an alternative Diagnosis Representation by the means of a ArgumentDataPrototype.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineOut/DcmDspStartRoutineOutSignal/DcmDspArgumentScaling/DcmDspAlternativeArgumentData
BSW Parameter	**BSW Type**
DcmDataElement	EcucForeignReferenceDef
BSW Description	
Alternative Diagnosis Representation for the data defined by the means of a ArgumentDataPrototype	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineOut/DcmDspStartRoutineOutSignal/DcmDspArgumentScaling
BSW Parameter	**BSW Type**
DcmDspAlternativeDataProps	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) of an alternative Diagnosis Representation by the means of ECU configuration parameters. The physical unit of the alternative data representation is defined by the DataPrototype referenced by DcmDspExternalSRDataElementClass. Additionally the definition of a text table mapping can be a defined for DcmDspDataTypeCategory TEXTTABLE and SCALE_LINEAR_AND_TEXTTABLE	
Template Description	

This class is a collection of properties relevant for data objects under various aspects. One could consider this class as a "pattern of inheritance by aggregation". The properties can be applied to all objects of all classes in which SwDataDefProps is aggregated. Note that not all of the attributes or associated elements are useful all of the time. Hence, the process definition (e.g. expressed with an OCL or a Document Control Instance MSR-DCI) has the task of implementing limitations. SwDataDefProps covers various aspects: * Structure of the data element for calibration use cases: is it a single value, a curve, or a map, but also the recordLayouts which specify how such elements are mapped/converted to the DataTypes in the programming language (or in AUTOSAR). This is mainly expressed by properties like swRecordLayout and swCalprmAxisSet * Implementation aspects, mainly expressed by swImplPolicy, swVariableAccessImplPolicy, swAddrMethod, swPointerTagetProps, baseType, implementationDataType and additionalNativeTypeQualifier * Access policy for the MCD system, mainly expressed by swCalibrationAccess * Semantics of the data element, mainly expressed by compuMethod and/or unit, dataConstr, invalidValue * Code generation policy provided by swRecordLayout
M2 Parameter
DataDictionary::DataDefProperties::SwDataDefProps

Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineOut/DcmDspStartRoutineOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps

BSW Parameter	**BSW Type**
DcmDspDataTypeCategory	EcucEnumerationParamDef
BSW Description	
Data category of the alternative Diagnosis Representation.	
Template Description	
The category is a keyword that specializes the semantics of the Identifiable. It affects the expected existence of attributes and the applicability of constraints.	
M2 Parameter	
GenericStructure::GeneralTemplateClasses::Identifiable::Identifiable.category	

Mapping Rule	**Mapping Type**
The value of the category of the compuMethod referenced by the swDataDefProps of the DiagnosticParameter with the role DiagnosticDataIdentifier.dataElement.	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineOut/DcmDspStartRoutineOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps

BSW Parameter	BSW Type
DcmDspLinearScale	EcucParamConfContainerDef

BSW Description
This container contains the configuration (parameters) of an linear scale of the alternative Diagnosis Representation.

Template Description

M2 Parameter

Mapping Rule	Mapping Type

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineOut/DcmDspStartRoutineOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps/DcmDspLinearScale

BSW Parameter	BSW Type
DcmDspDiagnosisRepresentationDataLowerRange	EcucFloatParamDef

BSW Description
Lower Range for this scale of the data in the alternative Diagnosis Representation.

Template Description
This specifies the lower limit of the constraint.

M2 Parameter
AsamHdo::Constraints::GlobalConstraints::PhysConstrs.lowerLimit

Mapping Rule	Mapping Type
PhysConstr referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticDataIdentifier.dataElement	full

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineOut/DcmDspStartRoutineOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps/DcmDspLinearScale

BSW Parameter	BSW Type
DcmDspDiagnosisRepresentationDataOffset	EcucFloatParamDef

BSW Description
Data offset of the alternative Diagnosis Representation for this scale.

Template Description
This is the numerator of the rational expression.

M2 Parameter
AsamHdo::ComputationMethod::CompuRationalCoeffs.compuNumerator

Mapping Rule	Mapping Type
first compuNumerator of compuMethod referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticDataIdentifier.dataElement	full

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineOut/DcmDspStartRoutineOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps/DcmDspLinearScale
BSW Parameter	**BSW Type**
DcmDspDiagnosisRepresentationDataResolution	EcucFloatParamDef
BSW Description	
Data resolution of the alternative Diagnosis Representation for this scale.	
Template Description	
This is the numerator of the rational expression.	
M2 Parameter	
AsamHdo::ComputationMethod::CompuRationalCoeffs.compuNumerator	
Mapping Rule	**Mapping Type**
The second compuNumerator in the scope of a CompuMethod referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticDataIdentifier.dataElement defines the scaling factor.	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineOut/DcmDspStartRoutineOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps/DcmDspLinearScale
BSW Parameter	**BSW Type**
DcmDspDiagnosisRepresentationDataUpperRange	EcucFloatParamDef
BSW Description	
Upper Range for this scale of the data in the alternative Diagnosis Representation.	
Template Description	
This specifies the upper limit of the constraint.	
M2 Parameter	
AsamHdo::Constraints::GlobalConstraints::PhysConstrs.upperLimit	
Mapping Rule	**Mapping Type**
PhysConstr referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticDataIdentifier.dataElement	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineOut/DcmDspStartRoutineOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps
BSW Parameter	**BSW Type**
DcmDspTextTableMapping	EcucParamConfContainerDef
BSW Description	

This container contains the configuration (parameters) of the mapping a DataPrototype typed by AutosarDataType that refer to a CompuMethods of category TEXTTABLE or SCALE_LINEAR_AND_TEXTTABLE. Each DcmDspTextTableMapping defines a value pair which is used to map the ECU internal value (DcmDspInternalDataValue) to the vale used in the diagnosis representation (DcmDspDiagnosisRepresentationDataValue) and vice versa. The set of all DcmDspTextTableMappings defines the whole mapping of data.
Template Description
This meta-class represents the ability to express the relationship between a physical value and the mathematical representation. Note that this is still independent of the technical implementation in data types. It only specifies the formula how the internal value corresponds to its physical pendant.
M2 Parameter
AsamHdo::ComputationMethod::CompuMethod

Mapping Rule	**Mapping Type**
This mapping applies if the CompuMethod.category is set to values TEXTTABLE or SCALE_LINEAR_AND_TEXTTABLE.	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineOut/DcmDspStartRoutineOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps/DcmDspTextTableMapping

BSW Parameter	**BSW Type**
DcmDspDiagnosisRepresentationDataValue	EcucIntegerParamDef
BSW Description	
The data value in the diagnosis representation.	
Template Description	
This represents a textual constant in the computation method.	
M2 Parameter	
AsamHdo::ComputationMethod::CompuConstTextContent.vt	

Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineOut/DcmDspStartRoutineOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps/DcmDspTextTableMapping

BSW Parameter	**BSW Type**
DcmDspInternalDataValue	EcucIntegerParamDef
BSW Description	
The ECU internal data value.	
Template Description	

CompuScale.lowerLimit: This specifies the lower limit of the scale. **CompuScale.upperLimit**: This specifies the upper limit of a of the scale.
M2 Parameter
AsamHdo::ComputationMethod::CompuScale.lowerLimit, AsamHdo::ComputationMethod::CompuScale.upperLimit

Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineOut/DcmDspStartRoutineOutSignal/DcmDspArgumentScaling

BSW Parameter	**BSW Type**
DcmDspAlternativeDataType	EcucParamConfContainerDef

BSW Description
This container contains the configuration (parameters) of an alternative Diagnosis Representation by the means of an ApplicationDataType. Additionally the definition of a text table mapping can be a defined for ApplicationDataTypes of category TEXTTABLE and SCALE_LINEAR_AND_TEXTTABLE.
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineOut/DcmDspStartRoutineOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataType

BSW Parameter	**BSW Type**
DcmApplicationDataType	EcucForeignReferenceDef

BSW Description
Alternative Diagnosis Representation for the data defined by the means of a ApplicationPrimitiveDataType of category VALUE or BOOLEAN.
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineOut/DcmDspStartRoutineOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataType

BSW Parameter	BSW Type
DcmDspTextTableMapping	EcucParamConfContainerDef

BSW Description

This container contains the configuration (parameters) of the mapping a DataPrototype typed by AutosarDataType that refer to a CompuMethods of category TEXTTABLE or SCALE_LINEAR_AND_TEXTTABLE.

Each DcmDspTextTableMapping defines a value pair which is used to map the ECU internal value (DcmDspInternalDataValue) to the vale used in the diagnosis representation (DcmDspDiagnosisRepresentationDataValue) and vice versa.

The set of all DcmDspTextTableMappings defines the whole mapping of data.

Template Description

This meta-class represents the ability to express the relationship between a physical value and the mathematical representation.

Note that this is still independent of the technical implementation in data types. It only specifies the formula how the internal value corresponds to its physical pendant.

M2 Parameter

AsamHdo::ComputationMethod::CompuMethod

Mapping Rule	Mapping Type
This mapping applies if the CompuMethod.category is set to values TEXTTABLE or SCALE_LINEAR_AND_TEXTTABLE.	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineOut/DcmDspStartRoutineOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataType/DcmDspTextTableMapping

BSW Parameter	BSW Type
DcmDspDiagnosisRepresentationDataValue	EcucIntegerParamDef

BSW Description

The data value in the diagnosis representation.

Template Description

This represents a textual constant in the computation method.

M2 Parameter

AsamHdo::ComputationMethod::CompuConstTextContent.vt

Mapping Rule	Mapping Type
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineOut/DcmDspStartRoutineOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataType/DcmDspTextTableMapping

BSW Parameter	BSW Type

DcmDspInternalDataValue	EcucIntegerParamDef
BSW Description	
The ECU internal data value.	
Template Description	
CompuScale.lowerLimit: This specifies the lower limit of the scale. **CompuScale.upperLimit**: This specifies the upper limit of a of the scale.	
M2 Parameter	
AsamHdo::ComputationMethod::CompuScale.lowerLimit, AsamHdo::ComputationMethod::CompuScale.upperLimit	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineOut/DcmDspStartRoutineOutSignal
BSW Parameter	**BSW Type**
DcmDspRoutineSignalEndianness	EcucEnumerationParamDef
BSW Description	
Defines the endianness of the data belonging to a Routine Out Signal for Start subfunction. If no DcmDspRoutineSignalEndianness is defined the value of DcmDspDataDefaultEndianness is applicable.	
Template Description	
This attribute specifies the byte order of the base type.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.byteOrder	
Mapping Rule	**Mapping Type**
baseType.baseTypeDefinition.byteOrder referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStartRoutine.response	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineOut/DcmDspStartRoutineOutSignal
BSW Parameter	**BSW Type**
DcmDspRoutineSignalLength	EcucIntegerParamDef
BSW Description	
Provide the length of the signal in the RoutineControl request/response. The length is defined in bits.	
Template Description	
The existence of this attribute turns the data instance into an array of data. The attribute determines the size of the array in terms of how many elements the array can take.	
M2 Parameter	
DiagnosticExtract::CommonDiagnostics::DiagnosticDataElement.maxNumberOfElements	
Mapping Rule	**Mapping Type**
Only in case of variable length required (according to constr_6008). Calculation: DcmDspRoutineSignalLength = maxNumberOfElements * 8.	full

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineOut/DcmDspStartRoutineOutSignal
BSW Parameter	**BSW Type**
DcmDspRoutineSignalPos	EcucIntegerParamDef
BSW Description	
Provide the position of the signal in the RoutineControl request/response. The position is defined in bits.	
Template Description	
This represents the bitOffset of the DiagnosticParameter	
M2 Parameter	
DiagnosticExtract::CommonDiagnostics::DiagnosticParameter.bitOffset	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineOut/DcmDspStartRoutineOutSignal
BSW Parameter	**BSW Type**
DcmDspRoutineSignalType	EcucEnumerationParamDef
BSW Description	
Provide the type of the signal in the RoutineControl request/response.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineOut/DcmDspStartRoutineOutSignal/DcmDspRoutineSignalType
BSW Parameter	**BSW Type**
BOOLEAN	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is boolean.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	

AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStartRoutine.response baseTypeEncoding = BOOLEAN baseTypeSize = 1	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineOut/DcmDspStartRoutineOutSignal/DcmDspRoutineSignalType
BSW Parameter	**BSW Type**
SINT16	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is sint16.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStartRoutine.response baseTypeEncoding = 2C baseTypeSize = 16	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineOut/DcmDspStartRoutineOutSignal/DcmDspRoutineSignalType
BSW Parameter	**BSW Type**
SINT32	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is sint32.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	

AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStartRoutine.response baseTypeEncoding = 2C baseTypeSize = 32	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDsp StartRoutineOut/DcmDspStartRoutineOutSignal/DcmDspRoutineSignalType
BSW Parameter	**BSW Type**
SINT8	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is sint8.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStartRoutine.response baseTypeEncoding = 2C baseTypeSize = 8	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDsp StartRoutineOut/DcmDspStartRoutineOutSignal/DcmDspRoutineSignalType
BSW Parameter	**BSW Type**
UINT16	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is uint16.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	

AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStartRoutine.response baseTypeEncoding = NONE baseTypeSize = 16	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineOut/DcmDspStartRoutineOutSignal/DcmDspRoutineSignalType
BSW Parameter	**BSW Type**
UINT32	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is uint32.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStartRoutine.response baseTypeEncoding = NONE baseTypeSize = 32	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineOut/DcmDspStartRoutineOutSignal/DcmDspRoutineSignalType
BSW Parameter	**BSW Type**
UINT8	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is uint8.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	

AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStartRoutine.response baseTypeEncoding = NONE baseTypeSize = 8	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStartRoutine/DcmDspStartRoutineOut/DcmDspStartRoutineOutSignal/DcmDspRoutineSignalType
BSW Parameter	**BSW Type**
VARIABLE_LENGTH	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is uint8[(DcmDspRoutineSignalLength+7)/8]. This is only valid for the last signal and when DcmDspRoutineSignalType is set to VARIABLE_LENGTH.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStartRoutine.response baseTypeEncoding = NONE baseTypeSize = 8 arraySizeSemantics = variableSize	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine
BSW Parameter	**BSW Type**
DcmDspStopRoutine	EcucParamConfContainerDef
BSW Description	
Provides the configuration of Stop subservice for RoutineControl service. Existence indicates that the StopRoutine in the RoutineControl is supported.	
Template Description	
This represents the ability to stop a running routine.	
M2 Parameter	
DiagnosticExtract::CommonDiagnostics::DiagnosticRoutine.stop	

Mapping Rule	Mapping Type
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine
BSW Parameter	**BSW Type**
DcmDspStopRoutineCommonAuthorizationRef	EcucReferenceDef
BSW Description	
Reference to DcmDspCommonAuthorization Common authorization configuration taken from the referenced DcmDspStopRoutineCommonAuthorizationRef. If there is no reference, no check on the commonly defined authorization conditions shall be done to stop the routine.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine
BSW Parameter	**BSW Type**
DcmDspStopRoutineFnc	EcucFunctionNameDef
BSW Description	
Function name for request to application to stop a routine. (Routine_Stop-function) This parameter is related to the interface Xxx_Stop.	
Template Description	
This is supposed to represent a reference to a BswServiceDependency. the latter is not derived from Referrable and therefore this detour needs to be implemented to still let BswServiceDependency become the target of a reference.	
M2 Parameter	
DiagnosticExtract::ServiceMapping::DiagnosticServiceSwMapping.mappedBswServiceDependency	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine
BSW Parameter	**BSW Type**
DcmDspStopRoutineIn	EcucParamConfContainerDef
BSW Description	
Provide description of input parameter of Stop subservice for RoutineControl service.	
Template Description	

This represents the request parameters.	
M2 Parameter	
DiagnosticExtract::CommonDiagnostics::DiagnosticStopRoutine.request	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineIn
BSW Parameter	**BSW Type**
DcmDspStopRoutineInSignal	EcucParamConfContainerDef
BSW Description	
Provide description of a routine signal used in RoutineControl service. The ordering defined via the index attribute of the subcontainers in this list represents the order of the dataInN elements in the XXX_Stop function call.	
Template Description	
This represents the related dataElement of the DiagnosticParameter	
M2 Parameter	
DiagnosticExtract::CommonDiagnostics::DiagnosticParameter.dataElement	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineIn/DcmDspStopRoutineInSignal
BSW Parameter	**BSW Type**
DcmDspArgumentScaling	EcucChoiceContainerDef
BSW Description	
This container contains the configuration (arguments) of an alternative Diagnosis Representation. Out if this the scaling between Diagnosis and ECU internal representation and vice versa can be calculated.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineIn/DcmDspStopRoutineInSignal/DcmDspArgumentScaling
BSW Parameter	**BSW Type**
DcmDspAlternativeArgumentData	EcucParamConfContainerDef

BSW Description	
This container contains the configuration (parameters) of an alternative Diagnosis Representation by the means of a ArgumentDataPrototype.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineIn/DcmDspStopRoutineInSignal/DcmDspArgumentScaling/DcmDspAlternativeArgumentData
BSW Parameter	**BSW Type**
DcmDataElement	EcucForeignReferenceDef
BSW Description	
Alternative Diagnosis Representation for the data defined by the means of a ArgumentDataPrototype	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineIn/DcmDspStopRoutineInSignal/DcmDspArgumentScaling
BSW Parameter	**BSW Type**
DcmDspAlternativeDataProps	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) of an alternative Diagnosis Representation by the means of ECU configuration parameters. The physical unit of the alternative data representation is defined by the DataPrototype referenced by DcmDspExternalSRDataElementClass. Additionally the definition of a text table mapping can be a defined for DcmDspDataTypeCategory TEXTTABLE and SCALE_LINEAR_AND_TEXTTABLE	
Template Description	

This class is a collection of properties relevant for data objects under various aspects. One could consider this class as a "pattern of inheritance by aggregation". The properties can be applied to all objects of all classes in which SwDataDefProps is aggregated. Note that not all of the attributes or associated elements are useful all of the time. Hence, the process definition (e.g. expressed with an OCL or a Document Control Instance MSR-DCI) has the task of implementing limitations. SwDataDefProps covers various aspects: * Structure of the data element for calibration use cases: is it a single value, a curve, or a map, but also the recordLayouts which specify how such elements are mapped/converted to the DataTypes in the programming language (or in AUTOSAR). This is mainly expressed by properties like swRecordLayout and swCalprmAxisSet * Implementation aspects, mainly expressed by swImplPolicy, swVariableAccessImplPolicy, swAddrMethod, swPointerTagetProps, baseType, implementationDataType and additionalNativeTypeQualifier * Access policy for the MCD system, mainly expressed by swCalibrationAccess * Semantics of the data element, mainly expressed by compuMethod and/or unit, dataConstr, invalidValue * Code generation policy provided by swRecordLayout
M2 Parameter
DataDictionary::DataDefProperties::SwDataDefProps

Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineIn/DcmDspStopRoutineInSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps

BSW Parameter	**BSW Type**
DcmDspDataTypeCategory	EcucEnumerationParamDef
BSW Description	
Data category of the alternative Diagnosis Representation.	
Template Description	
The category is a keyword that specializes the semantics of the Identifiable. It affects the expected existence of attributes and the applicability of constraints.	
M2 Parameter	
GenericStructure::GeneralTemplateClasses::Identifiable::Identifiable.category	

Mapping Rule	**Mapping Type**
The value of the category of the compuMethod referenced by the swDataDefProps of the DiagnosticParameter with the role DiagnosticDataIdentifier.dataElement.	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineIn/DcmDspStopRoutineInSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps

BSW Parameter	BSW Type
DcmDspLinearScale	EcucParamConfContainerDef

BSW Description
This container contains the configuration (parameters) of an linear scale of the alternative Diagnosis Representation.

Template Description

M2 Parameter

Mapping Rule	Mapping Type

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineIn/DcmDspStopRoutineInSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps/DcmDspLinearScale

BSW Parameter	BSW Type
DcmDspDiagnosisRepresentationDataLowerRange	EcucFloatParamDef

BSW Description
Lower Range for this scale of the data in the alternative Diagnosis Representation.

Template Description
This specifies the lower limit of the constraint.

M2 Parameter
AsamHdo::Constraints::GlobalConstraints::PhysConstrs.lowerLimit

Mapping Rule	Mapping Type
PhysConstr referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticDataIdentifier.dataElement	full

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineIn/DcmDspStopRoutineInSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps/DcmDspLinearScale

BSW Parameter	BSW Type
DcmDspDiagnosisRepresentationDataOffset	EcucFloatParamDef

BSW Description
Data offset of the alternative Diagnosis Representation for this scale.

Template Description
This is the numerator of the rational expression.

M2 Parameter
AsamHdo::ComputationMethod::CompuRationalCoeffs.compuNumerator

Mapping Rule	Mapping Type
first compuNumerator of compuMethod referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticDataIdentifier.dataElement	full

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineIn/DcmDspStopRoutineInSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps/DcmDspLinearScale
BSW Parameter	**BSW Type**
DcmDspDiagnosisRepresentationDataResolution	EcucFloatParamDef
BSW Description	
Data resolution of the alternative Diagnosis Representation for this scale.	
Template Description	
This is the numerator of the rational expression.	
M2 Parameter	
AsamHdo::ComputationMethod::CompuRationalCoeffs.compuNumerator	
Mapping Rule	**Mapping Type**
The second compuNumerator in the scope of a CompuMethod referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticDataIdentifier.dataElement defines the scaling factor.	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineIn/DcmDspStopRoutineInSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps/DcmDspLinearScale
BSW Parameter	**BSW Type**
DcmDspDiagnosisRepresentationDataUpperRange	EcucFloatParamDef
BSW Description	
Upper Range for this scale of the data in the alternative Diagnosis Representation.	
Template Description	
This specifies the upper limit of the constraint.	
M2 Parameter	
AsamHdo::Constraints::GlobalConstraints::PhysConstrs.upperLimit	
Mapping Rule	**Mapping Type**
PhysConstr referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticDataIdentifier.dataElement	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineIn/DcmDspStopRoutineInSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps
BSW Parameter	**BSW Type**
DcmDspTextTableMapping	EcucParamConfContainerDef
BSW Description	

This container contains the configuration (parameters) of the mapping a DataPrototype typed by AutosarDataType that refer to a CompuMethods of category TEXTTABLE or SCALE_LINEAR_AND_TEXTTABLE. Each DcmDspTextTableMapping defines a value pair which is used to map the ECU internal value (DcmDspInternalDataValue) to the vale used in the diagnosis representation (DcmDspDiagnosisRepresentationDataValue) and vice versa. The set of all DcmDspTextTableMappings defines the whole mapping of data.
Template Description
This meta-class represents the ability to express the relationship between a physical value and the mathematical representation. Note that this is still independent of the technical implementation in data types. It only specifies the formula how the internal value corresponds to its physical pendant.
M2 Parameter
AsamHdo::ComputationMethod::CompuMethod

Mapping Rule	**Mapping Type**
This mapping applies if the CompuMethod.category is set to values TEXTTABLE or SCALE_LINEAR_AND_TEXTTABLE.	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineIn/DcmDspStopRoutineInSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps/DcmDspTextTableMapping

BSW Parameter	**BSW Type**
DcmDspDiagnosisRepresentationDataValue	EcucIntegerParamDef
BSW Description	
The data value in the diagnosis representation.	
Template Description	
This represents a textual constant in the computation method.	
M2 Parameter	
AsamHdo::ComputationMethod::CompuConstTextContent.vt	

Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineIn/DcmDspStopRoutineInSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps/DcmDspTextTableMapping

BSW Parameter	**BSW Type**
DcmDspInternalDataValue	EcucIntegerParamDef
BSW Description	
The ECU internal data value.	
Template Description	

CompuScale.lowerLimit: This specifies the lower limit of the scale. **CompuScale.upperLimit**: This specifies the upper limit of a of the scale.
M2 Parameter
AsamHdo::ComputationMethod::CompuScale.lowerLimit, AsamHdo::ComputationMethod::CompuScale.upperLimit

Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineIn/DcmDspStopRoutineInSignal/DcmDspArgumentScaling

BSW Parameter	**BSW Type**
DcmDspAlternativeDataType	EcucParamConfContainerDef

BSW Description
This container contains the configuration (parameters) of an alternative Diagnosis Representation by the means of an ApplicationDataType. Additionally the definition of a text table mapping can be a defined for ApplicationDataTypes of category TEXTTABLE and SCALE_LINEAR_AND_TEXTTABLE.
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineIn/DcmDspStopRoutineInSignal/DcmDspArgumentScaling/DcmDspAlternativeDataType

BSW Parameter	**BSW Type**
DcmApplicationDataType	EcucForeignReferenceDef

BSW Description
Alternative Diagnosis Representation for the data defined by the means of a ApplicationPrimitiveDataType of category VALUE or BOOLEAN.
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineIn/DcmDspStopRoutineInSignal/DcmDspArgumentScaling/DcmDspAlternativeDataType

BSW Parameter	BSW Type
DcmDspTextTableMapping	EcucParamConfContainerDef

BSW Description

This container contains the configuration (parameters) of the mapping a DataPrototype typed by AutosarDataType that refer to a CompuMethods of category TEXTTABLE or SCALE_LINEAR_AND_TEXTTABLE.

Each DcmDspTextTableMapping defines a value pair which is used to map the ECU internal value (DcmDspInternalDataValue) to the vale used in the diagnosis representation (DcmDspDiagnosisRepresentationDataValue) and vice versa.

The set of all DcmDspTextTableMappings defines the whole mapping of data.

Template Description

This meta-class represents the ability to express the relationship between a physical value and the mathematical representation.

Note that this is still independent of the technical implementation in data types. It only specifies the formula how the internal value corresponds to its physical pendant.

M2 Parameter

AsamHdo::ComputationMethod::CompuMethod

Mapping Rule	Mapping Type
This mapping applies if the CompuMethod.category is set to values TEXTTABLE or SCALE_LINEAR_AND_TEXTTABLE.	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineIn/DcmDspStopRoutineInSignal/DcmDspArgumentScaling/DcmDspAlternativeDataType/DcmDspTextTableMapping

BSW Parameter	BSW Type
DcmDspDiagnosisRepresentationDataValue	EcucIntegerParamDef

BSW Description

The data value in the diagnosis representation.

Template Description

This represents a textual constant in the computation method.

M2 Parameter

AsamHdo::ComputationMethod::CompuConstTextContent.vt

Mapping Rule	Mapping Type
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineIn/DcmDspStopRoutineInSignal/DcmDspArgumentScaling/DcmDspAlternativeDataType/DcmDspTextTableMapping

BSW Parameter	BSW Type

DcmDspInternalDataValue	EcucIntegerParamDef
BSW Description	
The ECU internal data value.	
Template Description	
CompuScale.lowerLimit: This specifies the lower limit of the scale. **CompuScale.upperLimit**: This specifies the upper limit of a of the scale.	
M2 Parameter	
AsamHdo::ComputationMethod::CompuScale.lowerLimit, AsamHdo::ComputationMethod::CompuScale.upperLimit	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineIn/DcmDspStopRoutineInSignal
BSW Parameter	**BSW Type**
DcmDspRoutineSignalEndianness	EcucEnumerationParamDef
BSW Description	
Defines the endianness of the data belonging to a Routine In Signal for Stop subfunction. If no DcmDspRoutineSignalEndianness is defined the value of DcmDspDataDefaultEndianness is applicable.	
Template Description	
This attribute specifies the byte order of the base type.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.byteOrder	
Mapping Rule	**Mapping Type**
baseType.baseTypeDefinition.byteOrder referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStopRoutine.request	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineIn/DcmDspStopRoutineInSignal
BSW Parameter	**BSW Type**
DcmDspRoutineSignalLength	EcucIntegerParamDef
BSW Description	
Provide the length in bits of the signal in the RoutineControl request/response	
Template Description	
The existence of this attribute turns the data instance into an array of data. The attribute determines the size of the array in terms of how many elements the array can take.	
M2 Parameter	
DiagnosticExtract::CommonDiagnostics::DiagnosticDataElement.maxNumberOfElements	
Mapping Rule	**Mapping Type**
Only in case of variable length required (according to constr_6008). Calculation: DcmDspRoutineSignalLength = maxNumberOfElements * 8	full

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineIn/DcmDspStopRoutineInSignal
BSW Parameter	**BSW Type**
DcmDspRoutineSignalPos	EcucIntegerParamDef
BSW Description	
Provide the position of the signal in the RoutineControl request/response. The position is defined in bits.	
Template Description	
This represents the bitOffset of the DiagnosticParameter	
M2 Parameter	
DiagnosticExtract::CommonDiagnostics::DiagnosticParameter.bitOffset	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineIn/DcmDspStopRoutineInSignal
BSW Parameter	**BSW Type**
DcmDspRoutineSignalType	EcucEnumerationParamDef
BSW Description	
Provide the type of the signal in the RoutineControl request/response.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineIn/DcmDspStopRoutineInSignal/DcmDspRoutineSignalType
BSW Parameter	**BSW Type**
BOOLEAN	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is boolean.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	

AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStopRoutine.request baseTypeEncoding = BOOLEAN baseTypeSize = 1	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineIn/DcmDspStopRoutineInSignal/DcmDspRoutineSignalType
BSW Parameter	**BSW Type**
SINT16	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is sint16.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStopRoutine.request baseTypeEncoding = 2C baseTypeSize = 16	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineIn/DcmDspStopRoutineInSignal/DcmDspRoutineSignalType
BSW Parameter	**BSW Type**
SINT32	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is sint32.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	

AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStopRoutine.request baseTypeEncoding = 2C baseTypeSize = 32	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineIn/DcmDspStopRoutineInSignal/DcmDspRoutineSignalType
BSW Parameter	**BSW Type**
SINT8	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is sint8.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStopRoutine.request baseTypeEncoding = 2C baseTypeSize = 8	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineIn/DcmDspStopRoutineInSignal/DcmDspRoutineSignalType
BSW Parameter	**BSW Type**
UINT16	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is uint16.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	

AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStopRoutine.request baseTypeEncoding = NONE baseTypeSize = 16	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineIn/DcmDspStopRoutineInSignal/DcmDspRoutineSignalType
BSW Parameter	**BSW Type**
UINT32	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is uint32.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
baseTypeEncoding = NONE baseTypeSize = 32	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineIn/DcmDspStopRoutineInSignal/DcmDspRoutineSignalType
BSW Parameter	**BSW Type**
UINT8	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is uint8.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**

referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStopRoutine.request baseTypeEncoding = NONE baseTypeSize = 8	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineIn/DcmDspStopRoutineInSignal/DcmDspRoutineSignalType
BSW Parameter	**BSW Type**
VARIABLE_LENGTH	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is uint8[(DcmDspRoutineSignalLength+7)/8]. This is only valid for the last signal and when DcmDspRoutineSignalType is set to VARIABLE_LENGTH.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStopRoutine.request baseTypeEncoding = NONE baseTypeSize = 8 arraySizeSemantics = variableSize	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine
BSW Parameter	**BSW Type**
DcmDspStopRoutineOut	EcucParamConfContainerDef
BSW Description	
Provide description of output parameter of Stop subservice for RoutineControl service.	
Template Description	
This represents the respone parameters.	
M2 Parameter	
DiagnosticExtract::CommonDiagnostics::DiagnosticStopRoutine.response	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineOut

BSW Parameter	BSW Type
DcmDspStopRoutineOutSignal	EcucParamConfContainerDef

BSW Description

Provide description of a routine signal used in RoutineControl service.

The ordering defined via the index attribute of the subcontainers in this list represents the order of the dataOutN elements in the XXX_Stop function call.

Template Description

This represents the related dataElement of the DiagnosticParameter

M2 Parameter

DiagnosticExtract::CommonDiagnostics::DiagnosticParameter.dataElement

Mapping Rule	Mapping Type
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineOut/DcmDspStopRoutineOutSignal

BSW Parameter	BSW Type
DcmDspArgumentScaling	EcucChoiceContainerDef

BSW Description

This container contains the configuration (arguments) of an alternative Diagnosis Representation. Out if this the scaling between Diagnosis and ECU internal representation and vice versa can be calculated.

Template Description

M2 Parameter

Mapping Rule	Mapping Type
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineOut/DcmDspStopRoutineOutSignal/DcmDspArgumentScaling

BSW Parameter	BSW Type
DcmDspAlternativeArgumentData	EcucParamConfContainerDef

BSW Description

This container contains the configuration (parameters) of an alternative Diagnosis Representation by the means of a ArgumentDataPrototype.

Template Description

M2 Parameter

Mapping Rule	Mapping Type

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineOut/DcmDspStopRoutineOutSignal/DcmDspArgumentScaling/DcmDspAlternativeArgumentData

BSW Parameter	BSW Type
DcmDataElement	EcucForeignReferenceDef

BSW Description
Alternative Diagnosis Representation for the data defined by the means of a ArgumentDataPrototype

Template Description

M2 Parameter

Mapping Rule	Mapping Type
	local

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineOut/DcmDspStopRoutineOutSignal/DcmDspArgumentScaling

BSW Parameter	BSW Type
DcmDspAlternativeDataProps	EcucParamConfContainerDef

BSW Description
This container contains the configuration (parameters) of an alternative Diagnosis Representation by the means of ECU configuration parameters. The physical unit of the alternative data representation is defined by the DataPrototype referenced by DcmDspExternalSRDataElementClass. Additionally the definition of a text table mapping can be a defined for DcmDspDataTypeCategory TEXTTABLE and SCALE_LINEAR_AND_TEXTTABLE

Template Description

This class is a collection of properties relevant for data objects under various aspects. One could consider this class as a "pattern of inheritance by aggregation". The properties can be applied to all objects of all classes in which SwDataDefProps is aggregated. Note that not all of the attributes or associated elements are useful all of the time. Hence, the process definition (e.g. expressed with an OCL or a Document Control Instance MSR-DCI) has the task of implementing limitations. SwDataDefProps covers various aspects: * Structure of the data element for calibration use cases: is it a single value, a curve, or a map, but also the recordLayouts which specify how such elements are mapped/converted to the DataTypes in the programming language (or in AUTOSAR). This is mainly expressed by properties like swRecordLayout and swCalprmAxisSet * Implementation aspects, mainly expressed by swImplPolicy, swVariableAccessImplPolicy, swAddrMethod, swPointerTagetProps, baseType, implementationDataType and additionalNativeTypeQualifier * Access policy for the MCD system, mainly expressed by swCalibrationAccess * Semantics of the data element, mainly expressed by compuMethod and/or unit, dataConstr, invalidValue * Code generation policy provided by swRecordLayout
M2 Parameter
DataDictionary::DataDefProperties::SwDataDefProps

Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineOut/DcmDspStopRoutineOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps

BSW Parameter	**BSW Type**
DcmDspDataTypeCategory	EcucEnumerationParamDef
BSW Description	
Data category of the alternative Diagnosis Representation.	
Template Description	
The category is a keyword that specializes the semantics of the Identifiable. It affects the expected existence of attributes and the applicability of constraints.	
M2 Parameter	
GenericStructure::GeneralTemplateClasses::Identifiable::Identifiable.category	

Mapping Rule	**Mapping Type**
The value of the category of the compuMethod referenced by the swDataDefProps of the DiagnosticParameter with the role DiagnosticDataIdentifier.dataElement.	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineOut/DcmDspStopRoutineOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps

BSW Parameter	BSW Type
DcmDspLinearScale	EcucParamConfContainerDef

BSW Description
This container contains the configuration (parameters) of an linear scale of the alternative Diagnosis Representation.

Template Description

M2 Parameter

Mapping Rule	Mapping Type

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineOut/DcmDspStopRoutineOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps/DcmDspLinearScale

BSW Parameter	BSW Type
DcmDspDiagnosisRepresentationDataLowerRange	EcucFloatParamDef

BSW Description
Lower Range for this scale of the data in the alternative Diagnosis Representation.

Template Description
This specifies the lower limit of the constraint.

M2 Parameter
AsamHdo::Constraints::GlobalConstraints::PhysConstrs.lowerLimit

Mapping Rule	Mapping Type
PhysConstr referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticDataIdentifier.dataElement	full

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineOut/DcmDspStopRoutineOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps/DcmDspLinearScale

BSW Parameter	BSW Type
DcmDspDiagnosisRepresentationDataOffset	EcucFloatParamDef

BSW Description
Data offset of the alternative Diagnosis Representation for this scale.

Template Description
This is the numerator of the rational expression.

M2 Parameter
AsamHdo::ComputationMethod::CompuRationalCoeffs.compuNumerator

Mapping Rule	Mapping Type
first compuNumerator of compuMethod referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticDataIdentifier.dataElement	full

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineOut/DcmDspStopRoutineOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps/DcmDspLinearScale

BSW Parameter	BSW Type
DcmDspDiagnosisRepresentationDataResolution	EcucFloatParamDef

BSW Description
Data resolution of the alternative Diagnosis Representation for this scale.

Template Description
This is the numerator of the rational expression.

M2 Parameter
AsamHdo::ComputationMethod::CompuRationalCoeffs.compuNumerator

Mapping Rule	Mapping Type
The second compuNumerator in the scope of a CompuMethod referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticDataIdentifier.dataElement defines the scaling factor.	full

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineOut/DcmDspStopRoutineOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps/DcmDspLinearScale

BSW Parameter	BSW Type
DcmDspDiagnosisRepresentationDataUpperRange	EcucFloatParamDef

BSW Description
Upper Range for this scale of the data in the alternative Diagnosis Representation.

Template Description
This specifies the upper limit of the constraint.

M2 Parameter
AsamHdo::Constraints::GlobalConstraints::PhysConstrs.upperLimit

Mapping Rule	Mapping Type
PhysConstr referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticDataIdentifier.dataElement	full

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineOut/DcmDspStopRoutineOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps

BSW Parameter	BSW Type
DcmDspTextTableMapping	EcucParamConfContainerDef

BSW Description

This container contains the configuration (parameters) of the mapping a DataPrototype typed by AutosarDataType that refer to a CompuMethods of category TEXTTABLE or SCALE_LINEAR_AND_TEXTTABLE. Each DcmDspTextTableMapping defines a value pair which is used to map the ECU internal value (DcmDspInternalDataValue) to the vale used in the diagnosis representation (DcmDspDiagnosisRepresentationDataValue) and vice versa. The set of all DcmDspTextTableMappings defines the whole mapping of data.
Template Description
This meta-class represents the ability to express the relationship between a physical value and the mathematical representation. Note that this is still independent of the technical implementation in data types. It only specifies the formula how the internal value corresponds to its physical pendant.
M2 Parameter
AsamHdo::ComputationMethod::CompuMethod

Mapping Rule	**Mapping Type**
This mapping applies if the CompuMethod.category is set to values TEXTTABLE or SCALE_LINEAR_AND_TEXTTABLE.	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineOut/DcmDspStopRoutineOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps/DcmDspTextTableMapping

BSW Parameter	**BSW Type**
DcmDspDiagnosisRepresentationDataValue	EcucIntegerParamDef
BSW Description	
The data value in the diagnosis representation.	
Template Description	
This represents a textual constant in the computation method.	
M2 Parameter	
AsamHdo::ComputationMethod::CompuConstTextContent.vt	

Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineOut/DcmDspStopRoutineOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataProps/DcmDspTextTableMapping

BSW Parameter	**BSW Type**
DcmDspInternalDataValue	EcucIntegerParamDef
BSW Description	
The ECU internal data value.	
Template Description	

CompuScale.lowerLimit: This specifies the lower limit of the scale. **CompuScale.upperLimit**: This specifies the upper limit of a of the scale.
M2 Parameter
AsamHdo::ComputationMethod::CompuScale.lowerLimit, AsamHdo::ComputationMethod::CompuScale.upperLimit

Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineOut/DcmDspStopRoutineOutSignal/DcmDspArgumentScaling

BSW Parameter	**BSW Type**
DcmDspAlternativeDataType	EcucParamConfContainerDef

BSW Description
This container contains the configuration (parameters) of an alternative Diagnosis Representation by the means of an ApplicationDataType. Additionally the definition of a text table mapping can be a defined for ApplicationDataTypes of category TEXTTABLE and SCALE_LINEAR_AND_TEXTTABLE.
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineOut/DcmDspStopRoutineOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataType

BSW Parameter	**BSW Type**
DcmApplicationDataType	EcucForeignReferenceDef

BSW Description
Alternative Diagnosis Representation for the data defined by the means of a ApplicationPrimitiveDataType of category VALUE or BOOLEAN.
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineOut/DcmDspStopRoutineOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataType

BSW Parameter	BSW Type
DcmDspTextTableMapping	EcucParamConfContainerDef

BSW Description

This container contains the configuration (parameters) of the mapping a DataPrototype typed by AutosarDataType that refer to a CompuMethods of category TEXTTABLE or SCALE_LINEAR_AND_TEXTTABLE.

Each DcmDspTextTableMapping defines a value pair which is used to map the ECU internal value (DcmDspInternalDataValue) to the vale used in the diagnosis representation (DcmDspDiagnosisRepresentationDataValue) and vice versa.

The set of all DcmDspTextTableMappings defines the whole mapping of data.

Template Description

This meta-class represents the ability to express the relationship between a physical value and the mathematical representation.

Note that this is still independent of the technical implementation in data types. It only specifies the formula how the internal value corresponds to its physical pendant.

M2 Parameter

AsamHdo::ComputationMethod::CompuMethod

Mapping Rule	Mapping Type
This mapping applies if the CompuMethod.category is set to values TEXTTABLE or SCALE_LINEAR_AND_TEXTTABLE.	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineOut/DcmDspStopRoutineOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataType/DcmDspTextTableMapping

BSW Parameter	BSW Type
DcmDspDiagnosisRepresentationDataValue	EcucIntegerParamDef

BSW Description

The data value in the diagnosis representation.

Template Description

This represents a textual constant in the computation method.

M2 Parameter

AsamHdo::ComputationMethod::CompuConstTextContent.vt

Mapping Rule	Mapping Type
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineOut/DcmDspStopRoutineOutSignal/DcmDspArgumentScaling/DcmDspAlternativeDataType/DcmDspTextTableMapping

BSW Parameter	BSW Type

DcmDspInternalDataValue	EcucIntegerParamDef
BSW Description	
The ECU internal data value.	
Template Description	
CompuScale.lowerLimit: This specifies the lower limit of the scale. **CompuScale.upperLimit**: This specifies the upper limit of a of the scale.	
M2 Parameter	
AsamHdo::ComputationMethod::CompuScale.lowerLimit, AsamHdo::ComputationMethod::CompuScale.upperLimit	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineOut/DcmDspStopRoutineOutSignal
BSW Parameter	**BSW Type**
DcmDspRoutineSignalEndianness	EcucEnumerationParamDef
BSW Description	
Defines the endianness of the data belonging to a Routine Out Signal for Stop subfunction. If no DcmDspRoutineSignalEndianness is defined the value of DcmDspDataDefaultEndianness is applicable.	
Template Description	
This attribute specifies the byte order of the base type.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.byteOrder	
Mapping Rule	**Mapping Type**
baseType.baseTypeDefinition.byteOrder referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStopRoutine.response	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineOut/DcmDspStopRoutineOutSignal
BSW Parameter	**BSW Type**
DcmDspRoutineSignalLength	EcucIntegerParamDef
BSW Description	
Provide the length in bits of the signal in the RoutineControl request/response	
Template Description	
The existence of this attribute turns the data instance into an array of data. The attribute determines the size of the array in terms of how many elements the array can take.	
M2 Parameter	
DiagnosticExtract::CommonDiagnostics::DiagnosticDataElement.maxNumberOfElements	
Mapping Rule	**Mapping Type**
"Only in case of variable length required (according to constr_6008). Calculation: DcmDspRoutineSignalLength = maxNumberOfElements * 8	full

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineOut/DcmDspStopRoutineOutSignal
BSW Parameter	**BSW Type**
DcmDspRoutineSignalPos	EcucIntegerParamDef
BSW Description	
Provide the position of the signal in the RoutineControl request/response. The position is defined in bits.	
Template Description	
This represents the bitOffset of the DiagnosticParameter	
M2 Parameter	
DiagnosticExtract::CommonDiagnostics::DiagnosticParameter.bitOffset	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineOut/DcmDspStopRoutineOutSignal
BSW Parameter	**BSW Type**
DcmDspRoutineSignalType	EcucEnumerationParamDef
BSW Description	
Provide the type of the signal in the RoutineControl request/response.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineOut/DcmDspStopRoutineOutSignal/DcmDspRoutineSignalType
BSW Parameter	**BSW Type**
BOOLEAN	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is boolean.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	

AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStopRoutine.response baseTypeEncoding = BOOLEAN baseTypeSize = 1	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineOut/DcmDspStopRoutineOutSignal/DcmDspRoutineSignalType
BSW Parameter	**BSW Type**
SINT16	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is sint16.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStopRoutine.response baseTypeEncoding = 2C baseTypeSize = 16	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineOut/DcmDspStopRoutineOutSignal/DcmDspRoutineSignalType
BSW Parameter	**BSW Type**
SINT32	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is sint32.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	

AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStopRoutine.response baseTypeEncoding = 2C baseTypeSize = 32	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineOut/DcmDspStopRoutineOutSignal/DcmDspRoutineSignalType
BSW Parameter	**BSW Type**
SINT8	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is sint8.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStopRoutine.response baseTypeEncoding = 2C baseTypeSize = 8	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineOut/DcmDspStopRoutineOutSignal/DcmDspRoutineSignalType
BSW Parameter	**BSW Type**
UINT16	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is uint16.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	

AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStopRoutine.response baseTypeEncoding = NONE baseTypeSize = 16	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineOut/DcmDspStopRoutineOutSignal/DcmDspRoutineSignalType
BSW Parameter	**BSW Type**
UINT32	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is uint32.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStopRoutine.response baseTypeEncoding = NONE baseTypeSize = 32	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineOut/DcmDspStopRoutineOutSignal/DcmDspRoutineSignalType
BSW Parameter	**BSW Type**
UINT8	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is uint8.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	

AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStopRoutine.response baseTypeEncoding = NONE baseTypeSize = 8	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspRoutine/DcmDspStopRoutine/DcmDspStopRoutineOut/DcmDspStopRoutineOutSignal/DcmDspRoutineSignalType
BSW Parameter	**BSW Type**
VARIABLE_LENGTH	EcucEnumerationLiteralDef
BSW Description	
Type of the signal is uint8[(DcmDspRoutineSignalLength+7)/8]. This is only valid for the last signal and when DcmDspRoutineSignalType is set to VARIABLE_LENGTH.	
Template Description	
BaseTypeDirectDefinition.baseTypeEncoding: This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. **BaseTypeDirectDefinition.baseTypeSize**: Describes the length of the data type specified in the container in bits.	
M2 Parameter	
AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeEncoding, AsamHdo::BaseTypes::BaseTypeDirectDefinition.baseTypeSize	
Mapping Rule	**Mapping Type**
referenced by swDataDefProps of the DiagnosticParameter with the role DiagnosticStopRoutine.response baseTypeEncoding = NONE baseTypeSize = 8 arraySizeSemantics = variableSize	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp
BSW Parameter	**BSW Type**
DcmDspSecurity	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (DSP parameter) for security level configuration (per security level) Description This container contains Rows of DcmDspSecurityRow	
Template Description	

M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspSecurity
BSW Parameter	**BSW Type**
DcmDspSecurityRow	EcucParamConfContainerDef
BSW Description	
Definition of a single Row of configuration for security level configuration (per security level) The name of this container is used to define the name of the R-Port through which the DCM accesses the interface SecurityAccess_{SecurityLevel}. The R-Port is named SecurityAccess_{SecurityLevel} where {SecurityLevel} is the name of the container DcmDspSecurityRow. If there is no reference, no check of security level shall be done.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspSecurity/DcmDspSecurityRow
BSW Parameter	**BSW Type**
DcmDspSecurityADRSize	EcucIntegerParamDef
BSW Description	
Size in bytes of the AccessDataRecord used in GetSeed	
Template Description	
This represents the size of the AccessDataRecord used in GetSeed. Unit:byte.	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticSecurityLevel.accessDataRecordSize	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspSecurity/DcmDspSecurityRow
BSW Parameter	**BSW Type**
DcmDspSecurityAttemptCounterEnabled	EcucBooleanParamDef
BSW Description	
Allows to enable the external handling of the security attempt counter (e.g. to survive a reset of the ECU).	
Template Description	

M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspSecurity/DcmDspSecurityRow
BSW Parameter	**BSW Type**
DcmDspSecurityCompareKeyFnc	EcucFunctionNameDef
BSW Description	
Function name to request the result of a key comparison. Parameter is only relevant if DcmDspSecurityUsePort=="USE_ ASYNCH_FNC". This parameter is related to the interface Xxx_CompareKey.	
Template Description	
This is supposed to represent a reference to a BswServiceDependency. the latter is not derived from Referrable and therefore this detour needs to be implemented to still let BswServiceDependency become the target of a reference.	
M2 Parameter	
DiagnosticExtract::ServiceMapping::DiagnosticServiceSwMapping.mappedBswServiceDependency	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspSecurity/DcmDspSecurityRow
BSW Parameter	**BSW Type**
DcmDspSecurityDelayTime	EcucFloatParamDef
BSW Description	
Delay time after failed security access in seconds. This is started after DcmDspSecurityNumAttDelay number of failed security accesses. min: A negative value is not allowed.	
Template Description	
This represents the delay time after a failed security access. Unit: second.	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticSecurityLevel.securityDelayTime	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspSecurity/DcmDspSecurityRow
BSW Parameter	**BSW Type**

DcmDspSecurityDelayTimeOnBoot	EcucFloatParamDef
BSW Description	
Value of the delay timer in case of 'power on' in seconds. This delay indicates the time at ECU boot power-on time during which the Dcm does not accept a security access. min: A negative value is not allowed.	
Template Description	
Start delay timer on power on in seconds. This delay indicates the time at ECU boot power-on time where the Dcm remains in the default session and does not accept a security access.	
M2 Parameter	
DiagnosticExtract::DiagnosticCommonProps::DiagnosticCommonProps.securityDelayTimeOnBoot	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspSecurity/DcmDspSecurityRow
BSW Parameter	**BSW Type**
DcmDspSecurityGetAttemptCounterFnc	EcucFunctionNameDef
BSW Description	
Function name to request the value of an attempt counter. Parameter is only relevant if DcmDspSecurityUsePort=="USE_ASYNCH_FNC". This parameter is related to the interface Xxx_ GetSecurityAttemptCounter.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspSecurity/DcmDspSecurityRow
BSW Parameter	**BSW Type**
DcmDspSecurityGetSeedFnc	EcucFunctionNameDef
BSW Description	
Callout function name used to request a seed. Parameter is only relevant if DcmDspSecurityUsePort=="USE_ASYNCH_FNC". This parameter is related to the interface Xxx_GetSeed.	
Template Description	
This is supposed to represent a reference to a BswServiceDependency. the latter is not derived from Referrable and therefore this detour needs to be implemented to still let BswServiceDependency become the target of a reference.	
M2 Parameter	
DiagnosticExtract::ServiceMapping::DiagnosticServiceSwMapping.mappedBswServiceDependency	
Mapping Rule	**Mapping Type**

1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspSecurity/DcmDspSecurityRow
BSW Parameter	**BSW Type**
DcmDspSecurityKeySize	EcucIntegerParamDef
BSW Description	
size of the security key (in Bytes).	
Template Description	
This represents the size of the security key. Unit: byte.	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticSecurityLevel.keySize	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspSecurity/DcmDspSecurityRow
BSW Parameter	**BSW Type**
DcmDspSecurityLevel	EcucIntegerParamDef
BSW Description	
Value of Security level. The locked state cannot be configured explicitly. 1,2,3...63: configuration dependent - Conversion formula to calculate SecurityLevel out of tester requested SecurityAccessType parameter: SecurityLevel = (SecurityAccessType (requestSeed) + 1) / 2 Type: Dcm_SecLevelType	
Template Description	
This would be 0x01, 0x03, 0x05, ... The sendKey id can be computed by adding 1 to the requestSeedId	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticService::SecurityAccess::DiagnosticSecurityAccess.requestSeedId	
Mapping Rule	**Mapping Type**
DcmDspSecurityLevel=(requestSeedId+1)/2	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspSecurity/DcmDspSecurityRow
BSW Parameter	**BSW Type**
DcmDspSecurityNumAttDelay	EcucIntegerParamDef
BSW Description	

Number of failed security accesses after which the delay time is activated	
Template Description	
This represents the number of failed security accesses after which the delay time is activated.	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticSecurityLevel.numFailedSecurityAccess	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspSecurity/DcmDspSecurityRow
BSW Parameter	**BSW Type**
DcmDspSecuritySeedSize	EcucIntegerParamDef
BSW Description	
size of the security seed (in Bytes).	
Template Description	
This represents the size of the security seed. Unit: byte.	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticSecurityLevel.seedSize	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspSecurity/DcmDspSecurityRow
BSW Parameter	**BSW Type**
DcmDspSecuritySetAttemptCounterFnc	EcucFunctionNameDef
BSW Description	
Function name to set the value of an attempt counter. Parameter is only relevant if DcmDspSecurityUsePort=="USE_ASYNCH_FNC". This parameter is related to the interface Xxx_ SetSecurityAttemptCounter.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspSecurity/DcmDspSecurityRow
BSW Parameter	**BSW Type**
DcmDspSecurityUsePort	EcucEnumerationParamDef
BSW Description	
Defines which kind of interface shall be used for security access.	
Template Description	

M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspSecurity/DcmDspSecurityRow/DcmDspSecurityUsePort
BSW Parameter	**BSW Type**
USE_ASYNCH_CLIENT_SERVER	EcucEnumerationLiteralDef
BSW Description	
The DCM will access the data using an R-Port requiring a asynchronous ClientServertInterface SecurityAccess_{SecurityLevel}. The R-Port is described in DcmDspSecurityRow description.	
Template Description	
M2 Parameter	
DiagnosticExtract::ServiceMapping::DiagnosticServiceSwMapping.mappedSwcServiceDependency	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspSecurity/DcmDspSecurityRow/DcmDspSecurityUsePort
BSW Parameter	**BSW Type**
USE_ASYNCH_FNC	EcucEnumerationLiteralDef
BSW Description	
The DCM will access the data using the functions that are defined in the parameters DcmDspSecurityGetSeedFnc and DcmDspSecurityCompareKeyFnc as well as the functions defined in DcmDspSecurityGetAttemptCounterFnc and DcmDspSecuritySetAttemptCounterFnc, if enabled by the parameter DcmDspSecurityAttemptCounterEnabled. DCM_E_PENDING return is allowed and OpStatus is existing as IN parameter.	
Template Description	
This is supposed to represent a reference to a BswServiceDependency. the latter is not derived from Referrable and therefore this detour needs to be implemented to still let BswServiceDependency become the target of a reference.	
M2 Parameter	
DiagnosticExtract::ServiceMapping::DiagnosticServiceSwMapping.mappedBswServiceDependency	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp	
BSW Parameter		**BSW Type**
DcmDspSession		EcucParamConfContainerDef
BSW Description		
Parent container holding single rows to configure particular sessions		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspSession	
BSW Parameter		**BSW Type**
DcmDspSessionRow		EcucParamConfContainerDef
BSW Description		
This container holds all parameters needed to configure a single session		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspSession/DcmDspSessionRow	
BSW Parameter		**BSW Type**
DcmDspSessionForBoot		EcucEnumerationParamDef
BSW Description		
This parameter defines whether this diagnostic session allows to jump to Bootloader (OEM Bootloader or System Supplier Bootloader) and determines, from which unit the final response will be sent. If this diagnostic session doesn't allow to jump to Bootloader the value DCM_NO_BOOT shall be chosen.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context

Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspSession/DcmDspSessionRow/DcmDspSessionForBoot

BSW Parameter	**BSW Type**
DCM_NO_BOOT	EcucEnumerationLiteralDef
BSW Description	
This diagnostic session doesn't allow to jump to Bootloader.	
Template Description	
This diagnostic session doesn't allow to jump to Bootloader.	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticJumpToBootLoaderEnum.noBoot	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspSession/DcmDspSessionRow/DcmDspSessionForBoot

BSW Parameter	**BSW Type**
DCM_OEM_BOOT	EcucEnumerationLiteralDef
BSW Description	
This diagnostic session allows to jump to OEM Bootloader and bootloader sends final response.	
Template Description	
This diagnostic session allows to jump to OEM Bootloader.	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticJumpToBootLoaderEnum.oemBoot	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspSession/DcmDspSessionRow/DcmDspSessionForBoot

BSW Parameter	**BSW Type**
DCM_OEM_BOOT_RESPAPP	EcucEnumerationLiteralDef
BSW Description	
This diagnostic session allows to jump to OEM Bootloader and application sends final response.	
Template Description	
This diagnostic session allows to jump to OEM Bootloader.	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticJumpToBootLoaderEnum.oemBoot	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspSession/DcmDspSessionRow/DcmDspSessionForBoot

BSW Parameter	BSW Type
DCM_SYS_BOOT	EcucEnumerationLiteralDef
BSW Description	
This diagnostic session allows to jump to System Supplier Bootloader and bootloader sends final response.	
Template Description	
This diagnostic session allows to jump to System Supplier Bootloader.	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticJumpToBootLoaderEnum.systemSupplierBoot	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspSession/DcmDspSessionRow/DcmDspSessionForBoot
BSW Parameter	**BSW Type**
DCM_SYS_BOOT_RESPAPP	EcucEnumerationLiteralDef
BSW Description	
This diagnostic session allows to jump to System Supplier Bootloader and application sends final response.	
Template Description	
This diagnostic session allows to jump to System Supplier Bootloader.	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticJumpToBootLoaderEnum.systemSupplierBoot	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspSession/DcmDspSessionRow
BSW Parameter	**BSW Type**
DcmDspSessionLevel	EcucIntegerParamDef
BSW Description	
subFunction value of the DiagnosticSession. 0, 127 and all values above 127 are reserved by ISO	
Template Description	
This is the numerical identifier used to identify the DiagnosticSession in the scope of diagnostic workflow	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticSession.id	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context

Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspSession/DcmDspSessionRow	
BSW Parameter		**BSW Type**
DcmDspSessionP2ServerMax		EcucFloatParamDef
BSW Description		
This is the session value for P2ServerMax in seconds (per Session). The AUTOSAR configuration standard is to use SI units, so this parameter is defined as a float value in seconds. DCM configuration tools must convert this float value to the appropriate value format for the use in the software implementation of DCM. This value is reported to the tester within the response to the 'Session Control' service.		
Template Description		
This is the session value for P2ServerMax in seconds (per Session Control). The AUTOSAR configuration standard is to use SI units, so this parameter is defined as a float value in seconds.		
M2 Parameter		
DiagnosticExtract::Dcm::DiagnosticSession.p2ServerMax		
Mapping Rule		**Mapping Type**
1:1 mapping		full
Mapping Status		**Mapping ID**
valid		

BSW Module	**BSW Context**	
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspSession/DcmDspSessionRow	
BSW Parameter		**BSW Type**
DcmDspSessionP2StarServerMax		EcucFloatParamDef
BSW Description		
This is the session value for P2*ServerMax in seconds (per Session). The AUTOSAR configuration standard is to use SI units, so this parameter is defined as a float value in seconds. DCM configuration tools must convert this float value to the appropriate value format for the use in the software implementation of DCM. This value is reported to the tester within the response to the 'Session Control' service.		
Template Description		
This is the session value for P2*ServerMax in seconds (per Session Control). The AUTOSAR configuration standard is to use SI units, so this parameter is defined as a float value in seconds.		
M2 Parameter		
DiagnosticExtract::Dcm::DiagnosticSession.p2StarServerMax		
Mapping Rule		**Mapping Type**
1:1 mapping		full
Mapping Status		**Mapping ID**
valid		

BSW Module	**BSW Context**	
Dcm	Dcm/DcmConfigSet/DcmDsp	
BSW Parameter		**BSW Type**
DcmDspVehInfo		EcucParamConfContainerDef
BSW Description		
This container contains the configuration (parameters) for one single VehicleInfoType of service $09		
Template Description		
M2 Parameter		

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspVehInfo
BSW Parameter	**BSW Type**
DcmDspVehInfoData	EcucParamConfContainerDef
BSW Description	
Data Item of an InfoType; ShortName is post-fix of the port interface name.	
Template Description	
M2 Parameter	

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspVehInfo/DcmDspVehInfoData
BSW Parameter	**BSW Type**
DcmDspVehInfoDataOrder	EcucIntegerParamDef
BSW Description	
Defines the order of the data item in the InfoType; values: 0..255; first data item having the order number 0; the next 1 and so on. The configuration of order needs to be unique per InfoType.	
Template Description	
M2 Parameter	

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspVehInfo/DcmDspVehInfoData
BSW Parameter	**BSW Type**
DcmDspVehInfoDataReadFnc	EcucFunctionNameDef
BSW Description	
Callout function name for reading InfoType data item. Only required in case parameter 'DcmDspVehInfoDataUsePort' is set to 'false'	
Template Description	
M2 Parameter	

Mapping Rule	**Mapping Type**

Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspVehInfo/DcmDspVehInfoData
BSW Parameter	**BSW Type**
DcmDspVehInfoDataSize	EcucIntegerParamDef
BSW Description	
Size in bytes of the InfoType data item.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspVehInfo/DcmDspVehInfoData
BSW Parameter	**BSW Type**
DcmDspVehInfoDataUsePort	EcucBooleanParamDef
BSW Description	
When this parameter is set to true the DCM will access the Data using an R-Port requiring a PortInteface IInfotypeServices_{VehInfoData}. The R-Port is named InfotypeServices_{VehInfoData} where {VEHINFODATA} is the name of the container DcmDspVehInfoData. In that case, the DcmDspVehInfoDataReadFnc is ignored and the RTE APIs are used. When this parameter is set to false, the DCM calls the function defined in DcmDspVehInfo-DataReadFnc.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspVehInfo
BSW Parameter	**BSW Type**
DcmDspVehInfoInfoType	EcucIntegerParamDef
BSW Description	
value of InfoType. Within each DcmConfigSet all DcmDspVehInfoInfoType values shall be unique.	
Template Description	

M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmDspVehInfo
BSW Parameter	**BSW Type**
DcmDspVehInfoNODIProvResp	EcucBooleanParamDef
BSW Description	
Indicate the Dcm, which side is responsible to fill the number of data items (NODI), Dcm or the provider of the InfoType data. In case the responsibility is on provider side, only one DcmDspVehInfoData container is allowed. * true: Provider is responsible for providing the number of data items parameter * false or not existing: Dcm is responsible for providing the number of data items parameter	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp
BSW Parameter	**BSW Type**
DcmResponseToEcuReset	EcucEnumerationParamDef
BSW Description	
Defines the answer to EcuReset service should come: Before or after the reset.	
Template Description	
This attribute defines whether the response to the EcuReset service shall be transmitted before or after the actual reset.	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticService::EcuReset::DiagnosticEcuReset.respondToReset	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmResponseToEcuReset
BSW Parameter	**BSW Type**
AFTER_RESET	EcucEnumerationLiteralDef
BSW Description	
Answer to EcuReset service should come after the reset.	
Template Description	

Answer to EcuReset service should come after the reset.	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticService::EcuReset::DiagnosticResponseToEcuReset Enum.respondAfterReset	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmDsp/DcmResponseToEcuReset
BSW Parameter	**BSW Type**
BEFORE_RESET	EcucEnumerationLiteralDef
BSW Description	
Answer to EcuReset service should come before the reset.	
Template Description	
Answer to EcuReset service should come before the reset.	
M2 Parameter	
DiagnosticExtract::Dcm::DiagnosticService::EcuReset::DiagnosticResponseToEcuReset Enum.respondBeforeReset	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet
BSW Parameter	**BSW Type**
DcmPageBufferCfg	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) for Page Buffer handling	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmConfigSet/DcmPageBufferCfg
BSW Parameter	**BSW Type**
DcmPagedBufferEnabled	EcucBooleanParamDef
BSW Description	
Allow to enable or disable the Paged buffer mechanism. true = Paged buffer handling enabled false = Paged Buffer handling disabled	
Template Description	

M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmPageBufferCfg
BSW Parameter	**BSW Type**
DcmPagedBufferTimeout	EcucFloatParamDef
BSW Description	
Allow to configure the Timeout in seconds towards the application for filling the next page. This parameter is only relevant if the Paged Buffer handling is enabled. (DcmPagedBufferEnabled = TRUE) The AUTOSAR configuration standard is to use SI units, so this parameter is defined as float value in seconds. Dcm configuration tools must convert this float value to the appropriate value format for the use in the software implementation of Dcm. min: A negative value is not allowed. upperMultiplicity: Exactly one Timeout must be specified per configuration. lowerMultiplicity: Exactly one Timeout must be specified per configuration.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet
BSW Parameter	**BSW Type**
DcmProcessingConditions	EcucParamConfContainerDef
BSW Description	
This container contains the configuration for mode arbitration functionality of the Dcm	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmProcessingConditions
BSW Parameter	**BSW Type**
DcmModeCondition	EcucParamConfContainerDef
BSW Description	
This container contains the configuration of a mode condition or an environmental conditions which can be used as argument in DcmModeRules. One DcmModeCondition shall contain either one DcmSwcModeRef or one DcmBswModeRef or one DcmSwcSRDataElementRef. Please note that the Dcm acts as well as mode manager. Therefore the references DcmSwcModeRef or one DcmBswModeRef. might point to provided ModeDeclarationGroupPrototypes of the Dcm itself as well as to provided ModeDeclarationGroupPrototypes of other Bsw Modules or software components. In case of a configured DcmSwcModeRef or DcmBswModeRef only the DcmConditionType DCM_EQUALS or DCM_EQUALS_NOT are applicable. In case of DcmSwcSRDataElementRef all literals of DcmConditionType are possible.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmProcessingConditions/DcmModeCondition
BSW Parameter	**BSW Type**
DcmBswModeRef	EcucInstanceReferenceDef
BSW Description	
This parameter references a mode of a ModeDeclarationGroupPrototype provided by a Basic Software Module used for the condition. Please note that such ModeDeclarationGroupPrototype are owned by a Basic Software Module Description in the role providedModeGroup.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmProcessingConditions/DcmModeCondition
BSW Parameter	**BSW Type**
DcmConditionType	EcucEnumerationParamDef

BSW Description	
This parameter specifies what kind of comparison that is made for the evaluation of the mode condition.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmProcessingConditions/DcmModeCondition
BSW Parameter	**BSW Type**
DcmSwcModeRef	EcucInstanceReferenceDef
BSW Description	
This parameter references a mode in a particular mode request port of a software component that is used for the condition.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmProcessingConditions/DcmModeCondition
BSW Parameter	**BSW Type**
DcmSwcSRDataElementRef	EcucReferenceDef
BSW Description	
Reference to environmental conditions. It is possible to reference a S/R Receiver-Port to read physical values and compare (equal, greater, less,...) them with a configured value that is defined by DcmSwcSRDataElementValueRef.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmProcessingConditions/DcmModeCondition
BSW Parameter	**BSW Type**
DcmSwcSRDataElementValueRef	EcucForeignReferenceDef

BSW Description	
Reference to a constant specification defining the compare value for environmental condition.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmProcessingConditions
BSW Parameter	**BSW Type**
DcmModeRule	EcucParamConfContainerDef
BSW Description	
This container contains the configuration of a mode rule which represents a logical expression with DcmModeConditions or other DcmModeRules as arguments. All arguments are processed with the operator defined by DcmLogicalOperator, for instance: Argument_A AND Argument_B AND Argument_C	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmProcessingConditions/DcmModeRule
BSW Parameter	**BSW Type**
DcmArgumentRef	EcucChoiceReferenceDef
BSW Description	
This is a choice reference either to a mode condition or a an other mode rule serving as sub-expression.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmConfigSet/DcmProcessingConditions/DcmModeRule
BSW Parameter	**BSW Type**
DcmLogicalOperator	EcucEnumerationParamDef

BSW Description	
This parameter specifies the logical operator to be used in the logical expression. If the expression only consists of a single condition this parameter shall not be used.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context	
Dcm	Dcm/DcmConfigSet/DcmProcessingConditions/DcmModeRule	
BSW Parameter		**BSW Type**
DcmModeRuleNrcValue		EcucIntegerParamDef
BSW Description		
Optional parameter which defines the NRC to be sent in case the mode rule condition is not valid.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm	
BSW Parameter		**BSW Type**
DcmGeneral		EcucParamConfContainerDef
BSW Description		
This container contains the configuration (parameters) for Component wide parameters		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dcm	Dcm/DcmGeneral	
BSW Parameter		**BSW Type**
DcmDDDIDStorage		EcucBooleanParamDef
BSW Description		

This configuration switch defines, whether DDDID definition is stored non-volatile or not. true: DDDID are stored non-volatile false: DDDID are only maintained volatile
Template Description
This configuration switch defines whether DDDID definition is handled as non-volatile information or not.
M2 Parameter
DiagnosticExtract::Dcm::DiagnosticService::DynamicallyDefineDataIdentifier::DiagnosticDynamicallyDefineDataIdentifierClass.configurationHandling

Mapping Rule	**Mapping Type**
volatile -> false, nonVolatile -> true	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmGeneral

BSW Parameter	**BSW Type**
DcmDevErrorDetect	EcucBooleanParamDef

BSW Description
Switches the Default Error Tracer (Det) detection and notification ON or OFF. * true: enabled (ON). * false: disabled (OFF).
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmGeneral

BSW Parameter	**BSW Type**
DcmHeaderFileInclusion	EcucStringParamDef

BSW Description
Name of the header file(s) to be included by the Dcm module containing the used C-callback declarations.
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmGeneral

BSW Parameter	BSW Type
DcmRespondAllRequest	EcucBooleanParamDef
BSW Description	
If set to FALSE the Dcm will not respond to diagnostic request that contains a service ID which is in the range from 0x40 to 0x7F or in the range from 0xC0 to 0xFF (Response IDs).	
Template Description	
If set to FALSE the DCM will not respond to diagnostic request that contains a service ID which is in the range from 0x40 to 0x7F or in the range from 0xC0 to 0xFF (Response IDs).	
M2 Parameter	
DiagnosticExtract::DiagnosticCommonProps::DiagnosticCommonProps.responseOnAllRequestSids	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmGeneral
BSW Parameter	**BSW Type**
DcmTaskTime	EcucFloatParamDef
BSW Description	
Allow to configure the time for the periodic cyclic task. Please note: This configuration value shall be equal to the value in the RTE module. The AUTOSAR configuration standard is to use SI units, so this parameter is defined as float value in seconds. Dcm configuration tools must convert this float value to the appropriate value format for the use in the software implementation of Dcm. min: A negative value and zero is not allowed. upperMultiplicity: Exactly one TaskTime must be specified per configuration. lowerMultiplicity: Exactly one TaskTime must be specified per configuration.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dcm	Dcm/DcmGeneral
BSW Parameter	**BSW Type**
DcmVersionInfoApi	EcucBooleanParamDef
BSW Description	
Preprocessor switch to enable or disable the output Version info of the functionality.	
Template Description	

M2 Parameter

Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dcm	Dcm/DcmGeneral

BSW Parameter	**BSW Type**
DcmVinRef	EcucReferenceDef

BSW Description
Reference to the Did containing the VIN Information. This parameter is needed for function Dcm_GetVin
Template Description
This meta-class represents the ability to model a diagnostic data identifier (DID) that is fully specified regarding the payload at configuration-time.
M2 Parameter
DiagnosticExtract::CommonDiagnostics::DiagnosticDataIdentifier

Mapping Rule	**Mapping Type**
This reference shall only apply to a DiagnosticDataIdentifier where the attribute representsVin is set to true.	full
Mapping Status	**Mapping ID**
valid	

E.3 Dem

BSW Module	**BSW Context**
Dem	Dem

BSW Parameter	**BSW Type**
DemConfigSet	EcucParamConfContainerDef

BSW Description
This container contains the configuration parameters and sub containers of the Dem module supporting multiple configuration sets.
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet

BSW Parameter	**BSW Type**
DemComponent	EcucParamConfContainerDef
BSW Description	

This container configures the monitored components and system dependencies.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemComponent
BSW Parameter	**BSW Type**
DemComponentFailedCallbackFnc	EcucFunctionNameDef
BSW Description	
Specifies the function to be called on component failed status changes.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemComponent
BSW Parameter	**BSW Type**
DemComponentIgnoresPriority	EcucBooleanParamDef
BSW Description	
This configuration switch defines, whether the priority of events at this component shall be ignored.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemComponent
BSW Parameter	**BSW Type**
DemImmediateChildComponentRef	EcucReferenceDef
BSW Description	
Reference to all immediate children of the current component.	
Template Description	
M2 Parameter	

Mapping Rule	Mapping Type
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet
BSW Parameter	**BSW Type**
DemDTC	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) for DemUdsDTC.	
Template Description	
M2 Parameter	

Mapping Rule	Mapping Type
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemDTC
BSW Parameter	**BSW Type**
DemDTCAttributesRef	EcucReferenceDef
BSW Description	
This parameter defines the DTC Attributes associated with the DemDTC.	
Template Description	
M2 Parameter	

Mapping Rule	Mapping Type
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemDTC
BSW Parameter	**BSW Type**
DemDTCFunctionalUnit	EcucIntegerParamDef
BSW Description	
DTCFuncitonalUnit is a 1-byte value which identifies the corresponding basic vehicle / system function which reports the DTC. This parameter is necessary for the report of severity information. If this parameter is configured for no DTC, the Dem provides no DTC functional unit information.	
Template Description	
This attribute specifies a 1-byte value which identifies the corresponding basic vehicle / system function which reports the DTC. This parameter is necessary for the report of severity information.	
M2 Parameter	

DiagnosticExtract::Dem::DiagnosticTroubleCode::DiagnosticTroubleCodeUds.functionalUnit	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemDTC
BSW Parameter	**BSW Type**
DemDTCSeverity	EcucEnumerationParamDef
BSW Description	
DTC severity according to ISO 14229-1. This parameter depends on the automotive manufacturer. If it is not configured, the value is counted as 'no severity'. If this parameter is configured for no DTC, the Dem provides no DTC severity information.	
Template Description	
DTC severity according to ISO 14229-1.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticTroubleCode::DiagnosticTroubleCodeUds.severity	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemDTC
BSW Parameter	**BSW Type**
DemDtcValue	EcucIntegerParamDef
BSW Description	
Unique Diagnostic Trouble Code value for UDS (Range: 0x000000 and 0xFFFFFF are reserved for DTC groups by ISO 14229-1)	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemDTC
BSW Parameter	**BSW Type**
DemObdDTCRef	EcucReferenceDef
BSW Description	
This parameter defines the OBD DTC configuration associated with the DemDTC. It is allowed to have events without a OBD DTC.	
Template Description	

M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemDTC

BSW Parameter	**BSW Type**
DemWWHOBDDTCClass	EcucEnumerationParamDef

BSW Description
DTC Class according to ISO 14229-1 [2013 version]. This parameter depends on the automotive manufacturer. If it is not configured, the value is marked as 'unclassified'. If this parameter is configured for no DTC, the Dem provides no DTC WWHOBD class information.
Template Description
DTC severity according to ISO 14229-1.
M2 Parameter
DiagnosticExtract::Dem::DiagnosticTroubleCode::DiagnosticTroubleCodeUds.severity

Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet

BSW Parameter	**BSW Type**
DemDTCAttributes	EcucParamConfContainerDef

BSW Description
This container contains the configuration (parameters) for DemDTCAttributes.
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemDTCAttributes

BSW Parameter	**BSW Type**
DemAgingAllowed	EcucBooleanParamDef

BSW Description
Switch to allow aging/unlearning of the event or not. true: aging allowed false: aging not allowed

Template Description	
This represents the decision whether aging is allowed for this DiagnosticEvent.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticEvent::DiagnosticEvent.agingAllowed	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemDTCAttributes
BSW Parameter	**BSW Type**
DemAgingCycleCounterThreshold	EcucIntegerParamDef
BSW Description	
Number of aging cycles needed to unlearn/delete the event.	
Template Description	
Number of aging cycles needed to unlearn/delete the event.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticAging::DiagnosticAging.threshold	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemDTCAttributes
BSW Parameter	**BSW Type**
DemAgingCycleCounterThresholdForTFSLC	EcucIntegerParamDef
BSW Description	
Number of aging cycles needed to reset the testFailedSinceLastClear Bit.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemDTCAttributes
BSW Parameter	**BSW Type**
DemAgingCycleRef	EcucReferenceDef
BSW Description	
Reference to the cycle which is triggering the aging of the event.	
Template Description	
This represents the applicable aging cycle.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticAging::DiagnosticAging.agingCycle	
Mapping Rule	**Mapping Type**

1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemDTCAttributes
BSW Parameter	**BSW Type**
DemDTCPriority	EcucIntegerParamDef
BSW Description	
Priority of the event/dtc, in view of full event buffer. A lower value means higher priority.	
Template Description	
Priority of the event, in view of full event buffer. A lower value means higher priority.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticTroubleCode::DiagnosticTroubleCodeProps.priority	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemDTCAttributes
BSW Parameter	**BSW Type**
DemDTCSignificance	EcucEnumerationParamDef
BSW Description	
Significance of the event, which indicates additional information concerning fault classification and resolution. It can be mapped as Dem-internal data element. It shall be configured, if it is a part of event related data.	
Template Description	
Significance of the event, which indicates additional information concerning fault classification and resolution.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticTroubleCode::DiagnosticTroubleCodeProps.significance	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemDTCAttributes/DemDTCSignificance
BSW Parameter	**BSW Type**
DEM_EVENT_SIGNIFICANCE_OCCURRENCE	EcucEnumerationLiteralDef
BSW Description	
issue, which indicates additional information concerning insufficient system behavior	
Template Description	
Issue, which indicates additional information concerning insufficient system behavior.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticTroubleCode::DiagnosticSignificanceEnum.occurence	
Mapping Rule	**Mapping Type**

1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemDTCAttributes
BSW Parameter	**BSW Type**
DemEventMemoryEntryFdcThresholdStorageValue	EcucIntegerParamDef
BSW Description	
Threshold to allocate an event memory entry and to capture the Freeze Frame.	
Template Description	
Threshold to allocate an event memory entry and to capture the Freeze Frame. Unit: none (attribute represents a counter value).	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticTroubleCode::DiagnosticTroubleCodeProps.fdcThresholdStorageValue	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemDTCAttributes
BSW Parameter	**BSW Type**
DemExtendedDataClassRef	EcucReferenceDef
BSW Description	
This reference defines the link to an extended data class sampler.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemDTCAttributes
BSW Parameter	**BSW Type**
DemFreezeFrameClassRef	EcucReferenceDef
BSW Description	
These references define the links to a freeze frame class sampler.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context	
Dem	Dem/DemConfigSet/DemDTCAttributes	
BSW Parameter		**BSW Type**
DemFreezeFrameRecNumClassRef		EcucReferenceDef
BSW Description		
This parameter defines the list of dedicated freeze frame record numbers associated with the diagnostic event. These record numbers are assigned to the freeze frame records (instead of calculated record numbers). This parameter is only required for configured record numeration (refer to DemTypeOfFreeze-FrameRecordNumeration).		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemConfigSet/DemDTCAttributes	
BSW Parameter		**BSW Type**
DemImmediateNvStorage		EcucBooleanParamDef
BSW Description		
Switch to enable immediate storage triggering of an according event memory entriy persistently to NVRAM. true: immediate non-volatile storage triggering enabled false: immediate non-volatile storage triggering disabled		
Template Description		
Switch to enable immediate storage triggering of an according event memory entry persistently to NVRAM. true: immediate non-volatile storage triggering enabled false: immediate non-volatile storage triggering disabled		
M2 Parameter		
DiagnosticExtract::Dem::DiagnosticTroubleCode::DiagnosticTroubleCodeProps.immediateNvData Storage		
Mapping Rule		**Mapping Type**
1:1 mapping		full
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemConfigSet/DemDTCAttributes	
BSW Parameter		**BSW Type**
DemJ1939DTC_J1939NodeRef		EcucReferenceDef
BSW Description		
Reference to a J1939 Node		
Template Description		

M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemDTCAttributes
BSW Parameter	**BSW Type**
DemJ1939ExpandedFreezeFrameClassRef	EcucReferenceDef
BSW Description	
These references define the links to a J1939 freeze frame class sampler.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemDTCAttributes
BSW Parameter	**BSW Type**
DemJ1939FreezeFrameClassRef	EcucReferenceDef
BSW Description	
These references define the links to a J1939 freeze frame class sampler.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemDTCAttributes
BSW Parameter	**BSW Type**
DemMaxNumberFreezeFrameRecords	EcucIntegerParamDef
BSW Description	
This parameter defines the number of according freeze frame records, which can maximal be stored for this event. Therefore all these freeze frame records have the same freeze frame class. This parameter is only required for calculated record numeration (refer to DemTypeOfFreeze-FrameRecordNumeration).	
Template Description	
This attribute defines the number of according freeze frame records, which can maximal be stored for this event. Therefore all these freeze frame records have the same freeze frame class.	

M2 Parameter	
DiagnosticExtract::Dem::DiagnosticTroubleCode::DiagnosticTroubleCodeProps.maxNumberFreezeFrameRecords	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemDTCAttributes
BSW Parameter	**BSW Type**
DemMemoryDestinationRef	EcucChoiceReferenceDef
BSW Description	
The event destination assigns events to none, one or two origins. If no event destination is assigned to a specific event, the event is handled internally and is not visible externally to the Dcm. If more than one event destination is assigned to a specific event, the event can be present in the corresponding origins.	
Template Description	
The event destination assigns events to none, one or multiple origins.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticTroubleCode::DiagnosticTroubleCodeProps.memoryDestination	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemDTCAttributes
BSW Parameter	**BSW Type**
DemWWHOBDFreezeFrameClassRef	EcucReferenceDef
BSW Description	
This reference defines the link to a WWH-OBD freeze frame class sampler.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet
BSW Parameter	**BSW Type**
DemDebounceCounterBasedClass	EcucParamConfContainerDef
BSW Description	
This container contains the configuration of Debounce Counter Based Class	
Template Description	
M2 Parameter	

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemDebounceCounterBasedClass
BSW Parameter	**BSW Type**
DemDebounceBehavior	EcucEnumerationParamDef
BSW Description	
This parameter defines how the event debounce algorithm will behave, if a related enable condition is not fulfilled or ControlDTCSetting of the related event is disabled.	
Template Description	
This attribute defines how the event debounce algorithm will behave, if a related enable condition is not fulfilled or ControlDTCSetting of the related event is disabled.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticDebouncingAlgorithm::DiagnosticDebounceAlgorithmProps.debounceBehavior	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemDebounceCounterBasedClass/DemDebounceBehavior
BSW Parameter	**BSW Type**
DEM_DEBOUNCE_FREEZE	EcucEnumerationLiteralDef
BSW Description	
The event debounce counter will be frozen with the current value and will not change while a related enable condition is not fulfilled or ControlDTCSetting of the related event is disabled. After all related enable conditions are fulfilled and ControlDTCSetting of the related event is enabled again, the event qualification will continue with the next report of the event (i.e. SetEventStatus).	
Template Description	
The event debounce counter will be frozen with the current value and will not change while a related enable condition is not fulfilled or ControlDTCSetting of the related event is disabled. After all related enable conditions are fulfilled and ControlDTCSetting of the related event is enabled again, the event qualification will continue with the next report of the event (i.e. SetEventStatus).	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticDebouncingAlgorithm::DiagnosticDebounceBehaviorEnum.freeze	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemDebounceCounterBasedClass/DemDebounceBehavior
BSW Parameter	**BSW Type**

DEM_DEBOUNCE_RESET	EcucEnumerationLiteralDef
BSW Description	
The event debounce counter will be reset to initial value if a related enable condition is not fulfilled or ControlDTCSetting of the related event is disabled. The qualification of the event will be restarted with the next valid event report.	
Template Description	
The event debounce counter will be reset to initial value if a related enable condition is not fulfilled or ControlDTCSetting of the related event is disabled. The qualification of the event will be restarted with the next valid event report.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticDebouncingAlgorithm::DiagnosticDebounceBehavior Enum.reset	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemDebounceCounterBasedClass
BSW Parameter	**BSW Type**
DemDebounceCounterDecrementStepSize	EcucIntegerParamDef
BSW Description	
Defines the step size for decrementation of the internal debounce counter (PREPASSED).	
Template Description	
This value shall be taken to decrement the internal debounce counter.	
M2 Parameter	
CommonStructure::ServiceNeeds::DiagEventDebounceCounterBased.counterDecrementStepSize	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemDebounceCounterBasedClass
BSW Parameter	**BSW Type**
DemDebounceCounterFailedThreshold	EcucIntegerParamDef
BSW Description	
Defines the value of the internal debounce counter, which indicates the failed status.	
Template Description	
This value defines the event-specific limit that indicates the "failed" counter status.	
M2 Parameter	
CommonStructure::ServiceNeeds::DiagEventDebounceCounterBased.counterFailedThreshold	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemDebounceCounterBasedClass
BSW Parameter	**BSW Type**

DemDebounceCounterIncrementStepSize	EcucIntegerParamDef
BSW Description	
Defines the step size for incrementation of the internal debounce counter (PREFAILED).	
Template Description	
This value shall be taken to increment the internal debounce counter.	
M2 Parameter	
CommonStructure::ServiceNeeds::DiagEventDebounceCounterBased.counterIncrementStepSize	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemDebounceCounterBasedClass
BSW Parameter	**BSW Type**
DemDebounceCounterJumpDown	EcucBooleanParamDef
BSW Description	
Switch for the activation of Jump-Down. true: Jump-Down activated false: Jump-Down deactivated	
Template Description	
This value activates or deactivates the counter jump-down behavior.	
M2 Parameter	
CommonStructure::ServiceNeeds::DiagEventDebounceCounterBased.counterJumpDown	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemDebounceCounterBasedClass
BSW Parameter	**BSW Type**
DemDebounceCounterJumpDownValue	EcucIntegerParamDef
BSW Description	
Jump-Down value of the internal debounce counter which is taken as initialization value for the counter when the respective step-down occurs.	
Template Description	
This value represents the initial value of the internal debounce counter if the counting direction changes from incrementing to decrementing.	
M2 Parameter	
CommonStructure::ServiceNeeds::DiagEventDebounceCounterBased.counterJumpDownValue	
Mapping Rule	**Mapping Type**
	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemDebounceCounterBasedClass
BSW Parameter	**BSW Type**

DemDebounceCounterJumpUp	EcucBooleanParamDef
BSW Description	
Switch for the activation of Jump-Up. true: Jump-Up activated false: Jump-Up deactivated	
Template Description	
This value activates or deactivates the counter jump-up behavior.	
M2 Parameter	
CommonStructure::ServiceNeeds::DiagEventDebounceCounterBased.counterJumpUp	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemDebounceCounterBasedClass
BSW Parameter	**BSW Type**
DemDebounceCounterJumpUpValue	EcucIntegerParamDef
BSW Description	
Jump-Up value of the internal debounce counter which is taken as initialization value for the counter when the respective step-up occurs.	
Template Description	
This value represents the initial value of the internal debounce counter if the counting direction changes from decrementing to incrementing.	
M2 Parameter	
CommonStructure::ServiceNeeds::DiagEventDebounceCounterBased.counterJumpUpValue	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemDebounceCounterBasedClass
BSW Parameter	**BSW Type**
DemDebounceCounterPassedThreshold	EcucIntegerParamDef
BSW Description	
Defines the value of the internal debounce counter, which indicates the passed status.	
Template Description	
This value defines the event-specific limit that indicates the "passed" counter status.	
M2 Parameter	
CommonStructure::ServiceNeeds::DiagEventDebounceCounterBased.counterPassedThreshold	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemDebounceCounterBasedClass
BSW Parameter	**BSW Type**

DemDebounceCounterStorage	EcucBooleanParamDef
BSW Description	
Switch to store the debounce counter value non-volatile or not. true: debounce counter value shall be stored non-volatile false: debounce counter value is volatile	
Template Description	
Switch to store the debounce counter value non-volatile or not. true: debounce counter value shall be stored non-volatile false: debounce counter value is volatile	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticDebouncingAlgorithm::DiagnosticDebounceAlgorithm Props.debounceCounterStorage	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet
BSW Parameter	**BSW Type**
DemDebounceTimeBaseClass	EcucParamConfContainerDef
BSW Description	
This container contains the configuration of Debounce Counter Based Class	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemDebounceTimeBaseClass
BSW Parameter	**BSW Type**
DemDebounceBehavior	EcucEnumerationParamDef
BSW Description	
This parameter defines how the event debounce algorithm will behave, if a related enable condition is not fulfilled or ControlDTCSetting of the related event is disabled.	
Template Description	
This attribute defines how the event debounce algorithm will behave, if a related enable condition is not fulfilled or ControlDTCSetting of the related event is disabled.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticDebouncingAlgorithm::DiagnosticDebounceAlgorithm Props.debounceBehavior	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context	
Dem	Dem/DemConfigSet/DemDebounceTimeBaseClass/DemDebounceBehavior	
BSW Parameter		**BSW Type**
DEM_DEBOUNCE_FREEZE		EcucEnumerationLiteralDef
BSW Description		
The event debounce timer will be frozen with the current value and will not change while a related enable condition is not fulfilled or ControlDTCSetting of the related event is disabled. After all related enable conditions are fulfilled and ControlDTCSetting of the related event is enabled again, the event qualification will continue with the next report of the event (i.e. SetEventStatus).		
Template Description		
The event debounce counter will be frozen with the current value and will not change while a related enable condition is not fulfilled or ControlDTCSetting of the related event is disabled. After all related enable conditions are fulfilled and ControlDTCSetting of the related event is enabled again, the event qualification will continue with the next report of the event (i.e. SetEventStatus).		
M2 Parameter		
DiagnosticExtract::Dem::DiagnosticDebouncingAlgorithm::DiagnosticDebounceBehavior Enum.freeze		
Mapping Rule		**Mapping Type**
1:1 mapping		full
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemConfigSet/DemDebounceTimeBaseClass/DemDebounceBehavior	
BSW Parameter		**BSW Type**
DEM_DEBOUNCE_RESET		EcucEnumerationLiteralDef
BSW Description		
The event debounce timer will be reset to initial value if a related enable condition is not fulfilled or ControlDTCSetting of the related event is disabled. The qualification of the event will be restarted with the next valid event report.		
Template Description		
The event debounce counter will be reset to initial value if a related enable condition is not fulfilled or ControlDTCSetting of the related event is disabled. The qualification of the event will be restarted with the next valid event report.		
M2 Parameter		
DiagnosticExtract::Dem::DiagnosticDebouncingAlgorithm::DiagnosticDebounceBehavior Enum.reset		
Mapping Rule		**Mapping Type**
1:1 mapping		full
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemConfigSet/DemDebounceTimeBaseClass	
BSW Parameter		**BSW Type**
DemDebounceTimeFailedThreshold		EcucFloatParamDef
BSW Description		
Defines the time out duration for "Event Failed" qualification. The AUTOSAR configuration standard is to use SI units, so this parameter is defined as float value in seconds. Dem configuration tools must convert this float value to the appropriate value format for the use in the software implementation of Dem.		

Template Description	
This value represents the event-specific delay indicating the "failed" status.	
M2 Parameter	
CommonStructure::ServiceNeeds::DiagEventDebounceTimeBased.timeFailedThreshold	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemDebounceTimeBaseClass
BSW Parameter	**BSW Type**
DemDebounceTimePassedThreshold	EcucFloatParamDef
BSW Description	
Defines the time out duration for "Event Passed" qualification. The AUTOSAR configuration standard is to use SI units, so this parameter is defined as float value in seconds. Dem configuration tools must convert this float value to the appropriate value format for the use in the software implementation of Dem.	
Template Description	
This value represents the event-specific delay indicating the "passed" status.	
M2 Parameter	
CommonStructure::ServiceNeeds::DiagEventDebounceTimeBased.timePassedThreshold	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet
BSW Parameter	**BSW Type**
DemDtrs	EcucParamConfContainerDef
BSW Description	
This container holds the configuration of DTRs collection.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemDtrs
BSW Parameter	**BSW Type**
DemDtr	EcucParamConfContainerDef
BSW Description	
This container holds the configuration of one individual DTR.	
Template Description	

M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemDtrs/DemDtr
BSW Parameter	**BSW Type**
DemDtrCompuDenominator0	EcucFloatParamDef
BSW Description	
Part of the conversion between the binary representation and the physical meaning analogous to the SW-C Template conversion CompuRationalCoeffs with 2 numerator coefficients and 1 denominator coefficient in the direction compuInternalToPhys. This is the only one supported denominator value, a constant divisor. The value 0 is not allowed.	
Template Description	
M2 Parameter	

Mapping Rule	**Mapping Type**
	local
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemDtrs/DemDtr
BSW Parameter	**BSW Type**
DemDtrCompuNumerator0	EcucFloatParamDef
BSW Description	
Part of the conversion between the binary representation and the physical meaning analogous to the SW-C Template conversion CompuRationalCoeffs with 2 numerator coefficients and 1 denominator coefficient in the direction compuInternalToPhys. This is the first numerator value, which is multiplied with x^ 0, i.e., the offset.	
Template Description	
M2 Parameter	

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemDtrs/DemDtr
BSW Parameter	**BSW Type**
DemDtrCompuNumerator1	EcucFloatParamDef

BSW Description	
Part of the conversion between the binary representation and the physical meaning analogous to the SW-C Template conversion CompuRationalCoeffs with 2 numerator coefficients and 1 denominator coefficient in the direction compuInternalToPhys. This is the second numerator value, which is multiplied with xˆ 1, i.e., the factor.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemDtrs/DemDtr
BSW Parameter	**BSW Type**
DemDtrEventRef	EcucReferenceDef
BSW Description	
Reference to the DemEventParameter this DTR is related to. If the related event is not configured, the Dem cannot ensure consistency between the DTR and the event.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemDtrs/DemDtr
BSW Parameter	**BSW Type**
DemDtrId	EcucIntegerParamDef
BSW Description	
The index identifier value assigned to this DTR. The value is generated during the Dem configuration process.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemDtrs/DemDtr
BSW Parameter	**BSW Type**
DemDtrMid	EcucIntegerParamDef

BSW Description
The OBDMID of the DTR. The values 0x00, 0x20, 0x40, 0x60, 0x80, 0xA0, 0xC0, 0xE0 are reserved.
Template Description
M2 Parameter

Mapping Rule	Mapping Type
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemDtrs/DemDtr

BSW Parameter	BSW Type
DemDtrTid	EcucIntegerParamDef
BSW Description	
The OBDTID of the DTR.	
Template Description	
M2 Parameter	

Mapping Rule	Mapping Type
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemDtrs/DemDtr

BSW Parameter	BSW Type
DemDtrUasid	EcucIntegerParamDef
BSW Description	
The UaSId the DTR data shall be scaled to, and reported together with the rescaled DTR data.	
Template Description	
M2 Parameter	

Mapping Rule	Mapping Type
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemDtrs/DemDtr

BSW Parameter	BSW Type
DemDtrUpdateKind	EcucEnumerationParamDef
BSW Description	

Update conditions applied by the Dem to reports of DTR values. Only supported if a related Event is configured. If no related Event is configured, the Dem behaves as if DemDtrUpdateKind is configured to "DEM_DTR_UPDATE_ALWAYS".
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet

BSW Parameter	**BSW Type**
DemEventParameter	EcucParamConfContainerDef

BSW Description
This container contains the configuration (parameters) for events.
Template Description
Specifies the abstract needs on the configuration of the Diagnostic Event Manager for one diagnostic event. Its shortName can be regarded as a symbol identifying the diagnostic event from the viewpoint of the component or module which owns this element. In case the diagnostic event specifies a production error, the shortName shall be the name of the production error.
M2 Parameter
CommonStructure::ServiceNeeds::DiagnosticEventNeeds

Mapping Rule	**Mapping Type**
In case the owner of the DiagnosticEventNeeds is a BSW module then the Dem EventParameter.shortName = {capitalizedMip}_{ServiceDependency.symbolic NameProps.symbol}.	full
Mapping Status	**Mapping ID**
valid	up_Dem_00002

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemEventParameter

BSW Parameter	**BSW Type**
DemCallbackClearEventAllowed	EcucParamConfContainerDef

BSW Description
The presence of this container indicates that the Dem has access to a "ClearEventAllowed" callback. In case there is a DemCallbackClearEventAllowedFnc, this parameter defines the name of the function that the Dem will call. In case there is no DemCallbackClearEventAllowedFnc, the Dem will have an R-Port requiring the interface CallbackClearEventAllowed whose name is generated by using the unique callback-prefix followed by the event name.
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**

Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemEventParameter/DemCallbackClearEventAllowed
BSW Parameter	**BSW Type**
DemCallbackClearEventAllowedFnc	EcucFunctionNameDef
BSW Description	
Function name of prototype "ClearEventAllowed".	
Template Description	
This attribute defines whether the Dem has access to a "ClearEventAllowed" callback.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticEvent::DiagnosticEvent.eventClearAllowed	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemEventParameter/DemCallbackClearEventAllowed
BSW Parameter	**BSW Type**
DemClearEventAllowedBehavior	EcucEnumerationParamDef
BSW Description	
Defines the resulting UDS status byte for the related event, which must not be cleared according to the ClearEventAllowed callback.	
Template Description	
This attribute defines the resulting UDS status byte for the related event, which shall not be cleared according to the ClearEventAllowed callback.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticEvent::DiagnosticEvent.clearEventBehavior	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemEventParameter/DemCallbackClearEventAllowed/ DemClearEventAllowedBehavior
BSW Parameter	**BSW Type**
DEM_ONLY_THIS_CYCLE_AND_READINESS	EcucEnumerationLiteralDef
BSW Description	
The <...>ThisOperationCycle and readiness bits of the event status byte are reset.	
Template Description	
The OperationCycle and readiness bits of the event status byte are reset.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticEvent::DiagnosticClearEventBehaviorEnum.onlyThisCycleAnd Readiness	
Mapping Rule	**Mapping Type**
1:1 mapping	full

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemEventParameter
BSW Parameter	**BSW Type**
DemCallbackEventDataChanged	EcucParamConfContainerDef
BSW Description	
The presence of this container indicates that the Dem has access to an "EventDataChanged" callback. In case there is a DemCallbackEventDataChangedFnc, this parameter defines the name of the function that the Dem will call. In case there is no DemCallbackEventDataChangedFnc, the Dem will have an R-Port requiring the interface CallbackEventDataChanged whose name is generated by using the unique callback-prefix followed by the event name.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemEventParameter/DemCallbackEventDataChanged
BSW Parameter	**BSW Type**
DemCallbackEventDataChangedFnc	EcucFunctionNameDef
BSW Description	
Function name of prototype "EventDataChanged"	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemEventParameter
BSW Parameter	**BSW Type**
DemCallbackEventStatusChanged	EcucParamConfContainerDef
BSW Description	

The presence of this container indicates, that the Dem has access to an "EventStatusChanged" callback, which the Dem will call to notify other components about the change in the status of an event. In case there is a DemCallbackEvenStatusChangedFnc, this parameter defines the name of the function that the Dem will call. In case there is no DemCallbackEvenStatusChangedFnc, the Dem will have an R-Port requiring the interface CallbackEventStatusChanged, whose name is generated by using the unique callback-prefix followed by the event name.
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemEventParameter/DemCallbackEventStatusChanged

BSW Parameter	**BSW Type**
DemCallbackEventStatusChangedFnc	EcucFunctionNameDef

BSW Description
Function name of prototype "EventStatusChanged"
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemEventParameter

BSW Parameter	**BSW Type**
DemCallbackInitMForE	EcucParamConfContainerDef

BSW Description
The presence of this container indicates, that the Dem has access to an "InitMonitorForEvent" callback, which the Dem will call to initialize a monitor. In case the container has a DemCallbackInitMForEFnc, this parameter defines the name of the function that the Dem will call. In case there is no DemCallbackInitMForEFnc, the Dem will have an R-Port requiring the interface CallbackInitMonitorForEvent, whose name is generated by using the unique callback-prefix followed by the event name.
Template Description
M2 Parameter

Mapping Rule	Mapping Type
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemEventParameter/DemCallbackInitMForE
BSW Parameter	**BSW Type**
DemCallbackInitMForEFnc	EcucFunctionNameDef
BSW Description	
Function name of prototype "InitMonitorForEvent".	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemEventParameter
BSW Parameter	**BSW Type**
DemComponentClassRef	EcucReferenceDef
BSW Description	
Reference to the monitored component.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemEventParameter
BSW Parameter	**BSW Type**
DemComponentPriority	EcucIntegerParamDef
BSW Description	
Specifies the priority within the component. A lower value means higher priority.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemEventParameter
BSW Parameter	**BSW Type**
DemDTCRef	EcucReferenceDef
BSW Description	
This parameter defines the DTC configuration (typically Uds) associated with the diagnostic event. It is allowed to have events without a DTC (e.g. for ECU-internal events triggering safety reactions without being reported via diagnostic communication). The same DemDTCAttributes can be used from several events, to combine these (refer to chapter "Combination of diagnostic event").	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemEventParameter
BSW Parameter	**BSW Type**
DemDebounceAlgorithmClass	EcucChoiceContainerDef
BSW Description	
Debounce algorithm class: counter based, time based, or monitor internal.	
Template Description	
This class represents the ability to specify the pre-debounce algorithm which is selected and/or required by the particular monitor. This class inherits from Identifiable in order to allow further documentation of the expected or implemented debouncing and to use the category for the identification of the expected / implemented debouncing.	
M2 Parameter	
CommonStructure::ServiceNeeds::DiagEventDebounceAlgorithm	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemEventParameter/DemDebounceAlgorithmClass
BSW Parameter	**BSW Type**
DemDebounceCounterBased	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) for counter based debouncing.	
Template Description	
This meta-class represents the ability to indicate that the counter-based debounce algorithm shall be used by the DEM for this diagnostic monitor. This is related to set the ECUC choice container DemDebounceAlgorithmClass to DemDebounceCounterBased.	
M2 Parameter	

CommonStructure::ServiceNeeds::DiagEventDebounceCounterBased	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemEventParameter/DemDebounceAlgorithmClass/DemDebounceCounterBased

BSW Parameter	**BSW Type**
DemDebounceCounterBasedClassRef	EcucReferenceDef
BSW Description	
This reference selects the DemDebounceCounterBasedClass applied for the debouncing of the DemEventParameter.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemEventParameter/DemDebounceAlgorithmClass

BSW Parameter	**BSW Type**
DemDebounceMonitorInternal	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) for monitor internal debouncing.	
Template Description	
This meta-class represents the ability to indicate that the pre-debounce algorithm shall be used by the DEM for this diagnostic monitor. This is related to setting the ECUC choice container DemDebounceAlgorithmClass to DemDebounceMonitorInternal. If the FaultDetectionAlogrithm is already known to be implemented by a specific BswModuleEntry the reference bswModuleEntry points to the function specification. If the FaultDetectionCounter value is accessible at a PortPrototype this PortPrototype shall be referenced by an assignedPort.	
M2 Parameter	
CommonStructure::ServiceNeeds::DiagEventDebounceMonitorInternal	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemEventParameter/DemDebounceAlgorithmClass/DemDebounceMonitorInternal

BSW Parameter	BSW Type
DemCallbackGetFDC	EcucParamConfContainerDef
BSW Description	
DemCallbackGetFDC specifies the callback (parameter DemCallbackGetFDCFnc is present) or R-Port (no parameter DemCallbackGetFDCFnc is present) to retrieve the fault detection counter value. In case no container is configured, no fault detection counter will be available.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemEventParameter/DemDebounceAlgorithmClass/DemDebounceMonitorInternal/DemCallbackGetFDC
BSW Parameter	**BSW Type**
DemCallbackGetFDCFnc	EcucFunctionNameDef
BSW Description	
This parameter defines the name of the function that the Dem will call to retrieve the fault-detection counter value from a complex driver.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemEventParameter/DemDebounceAlgorithmClass
BSW Parameter	**BSW Type**
DemDebounceTimeBase	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) for time based debouncing.	
Template Description	
This meta-class represents the ability to indicate that the time-based pre-debounce algorithm shall be used by the DEM for this diagnostic monitor. This is related to set the ECUC choice container DemDebounceAlgorithmClass to DemDebounceTimeBase.	
M2 Parameter	
CommonStructure::ServiceNeeds::DiagEventDebounceTimeBased	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemEventParameter/DemDebounceAlgorithmClass/DemDebounceTimeBase

BSW Parameter	BSW Type
DemDebounceTimeBaseRef	EcucReferenceDef

BSW Description
This reference selects the DemDebounceTimeBaseClass applied for the debouncing of the DemEventParameter.

Template Description

M2 Parameter

Mapping Rule	Mapping Type

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemEventParameter

BSW Parameter	BSW Type
DemEnableConditionGroupRef	EcucReferenceDef

BSW Description
References an enable condition group.

Template Description

M2 Parameter

Mapping Rule	Mapping Type

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemEventParameter

BSW Parameter	BSW Type
DemEventAvailable	EcucBooleanParamDef

BSW Description
This parameter configures an Event as unavailable. It is treated by Dem as if it does not exist. true = Event is available false = Event is not available

Template Description

M2 Parameter

Mapping Rule	Mapping Type

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context	
Dem	Dem/DemConfigSet/DemEventParameter	
BSW Parameter		**BSW Type**
DemEventFailureCycleCounterThreshold		EcucIntegerParamDef
BSW Description		
Defines the number of failure cycles for the event based fault confirmation. If this parameter is enabled, fault confirmation of the event is enabled accordingly.		
Template Description		
This attribute defines the number of failure cycles for the event based fault confirmation.		
M2 Parameter		
DiagnosticExtract::Dem::DiagnosticEvent::DiagnosticEvent.eventFailureCycleCounterThreshold		
Mapping Rule		**Mapping Type**
1:1 mapping		full
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemConfigSet/DemEventParameter	
BSW Parameter		**BSW Type**
DemEventId		EcucIntegerParamDef
BSW Description		
Unique identifier of a diagnostic event. This parameter should not be changeable by user, because the Id should be generated by Dem itself to prevent gaps and multiple use of an Id. The events should be sequentially ordered beginning with 1 and no gaps in between.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemConfigSet/DemEventParameter	
BSW Parameter		**BSW Type**
DemEventKind		EcucEnumerationParamDef
BSW Description		
This parameter is used to distinguish between SW-C and BSW events. SW-C events are reported by Dem_SetEventStatus API and BSW events are reported by Dem_ReportErrorStatus API.		
Template Description		
This attribute is used to distinguish between SWC and BSW events.		
M2 Parameter		
DiagnosticExtract::Dem::DiagnosticEvent::DiagnosticEvent.eventKind		
Mapping Rule		**Mapping Type**
1:1 mapping		full
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemConfigSet/DemEventParameter/DemEventKind	
BSW Parameter		**BSW Type**
DEM_EVENT_KIND_BSW		EcucEnumerationLiteralDef
BSW Description		
The event is a assigned to a BSW module		
Template Description		
The event is assigned to a BSW module.		
M2 Parameter		
DiagnosticExtract::Dem::DiagnosticEvent::DiagnosticEventKindEnum.bsw		
Mapping Rule		**Mapping Type**
1:1 mapping		full
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemConfigSet/DemEventParameter/DemEventKind	
BSW Parameter		**BSW Type**
DEM_EVENT_KIND_SWC		EcucEnumerationLiteralDef
BSW Description		
The event is a assigned to a SW-C		
Template Description		
The event is assigned to a SWC.		
M2 Parameter		
DiagnosticExtract::Dem::DiagnosticEvent::DiagnosticEventKindEnum.swc		
Mapping Rule		**Mapping Type**
1:1 mapping		full
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemConfigSet/DemEventParameter	
BSW Parameter		**BSW Type**
DemFFPrestorageSupported		EcucBooleanParamDef
BSW Description		
If this parameter is set to true, then the Prestorage of FreezeFrames is supported by the assigned event. This parameter is useful to calculate the buffer size.		
Template Description		
This attribute describes whether the Prestorage of FreezeFrames is supported by the assigned event or not. True: Prestorage of FreezeFrames is supported False: Prestorage of FreezeFrames is not supported		
M2 Parameter		
DiagnosticExtract::Dem::DiagnosticEvent::DiagnosticEvent.prestorageFreezeFrame		
Mapping Rule		**Mapping Type**
1:1 mapping		full
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemEventParameter
BSW Parameter	**BSW Type**
DemIndicatorAttribute	EcucParamConfContainerDef
BSW Description	
This container contains the event specific configuration of Indicators.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemEventParameter/DemIndicatorAttribute
BSW Parameter	**BSW Type**
DemIndicatorBehaviour	EcucEnumerationParamDef
BSW Description	
Behaviour of the linked indicator	
Template Description	
Behavior of the linked indicator.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticEvent::DiagnosticConnectedIndicator.behavior	
Mapping Rule	**Mapping Type**
	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemEventParameter/DemIndicatorAttribute/DemIndicator Behaviour
BSW Parameter	**BSW Type**
DEM_INDICATOR_BLINKING	EcucEnumerationLiteralDef
BSW Description	
The indicator blinks when the event has status FAILED Not relevant with J1939.	
Template Description	
The indicator blinks when the event has status FAILED.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticEvent::DiagnosticConnectedIndicatorBehaviorEnum.blinkMode	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemEventParameter/DemIndicatorAttribute/DemIndicator Behaviour

BSW Parameter	BSW Type
DEM_INDICATOR_BLINK_CONT	EcucEnumerationLiteralDef
BSW Description	
The indicator is active and blinks when the event has status FAILED Not relevant with J1939.	
Template Description	
The indicator is active and blinks when the event has status FAILED.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticEvent::DiagnosticConnectedIndicatorBehaviorEnum.blinkOr ContinuousOnMode	
Mapping Rule	**Mapping Type**
	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemEventParameter/DemIndicatorAttribute/DemIndicator Behaviour
BSW Parameter	**BSW Type**
DEM_INDICATOR_CONTINUOUS	EcucEnumerationLiteralDef
BSW Description	
The indicator is active when the even has status FAILED	
Template Description	
The indicator is active when the event has status FAILED.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticEvent::DiagnosticConnectedIndicatorBehaviorEnum.continuous OnMode	
Mapping Rule	**Mapping Type**
	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemEventParameter/DemIndicatorAttribute/DemIndicator Behaviour
BSW Parameter	**BSW Type**
DEM_INDICATOR_FAST_FLASH	EcucEnumerationLiteralDef
BSW Description	
Flash Indicator Lamp should be set to ’Fast Flash’	
Template Description	
Flash Indicator Lamp should be set to "Fast Flash".	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticEvent::DiagnosticConnectedIndicatorBehaviorEnum.fastFlash-ingMode	
Mapping Rule	**Mapping Type**
	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context

Dem	Dem/DemConfigSet/DemEventParameter/DemIndicatorAttribute/DemIndicatorBehaviour

BSW Parameter	**BSW Type**
DEM_INDICATOR_SLOW_FLASH	EcucEnumerationLiteralDef
BSW Description	
Flash Indicator Lamp should be set to 'Slow Flash'	
Template Description	
Flash Indicator Lamp should be set to "Slow Flash".	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticEvent::DiagnosticConnectedIndicatorBehaviorEnum.slowFlashingMode	
Mapping Rule	**Mapping Type**
	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemEventParameter/DemIndicatorAttribute
BSW Parameter	**BSW Type**
DemIndicatorFailureCycleCounterThreshold	EcucIntegerParamDef
BSW Description	
Defines the number of failure cycles for the WarningIndicatorOnCriteria.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemEventParameter/DemIndicatorAttribute
BSW Parameter	**BSW Type**
DemIndicatorHealingCycleCounterThreshold	EcucIntegerParamDef
BSW Description	
Defines the number of healing cycles for the WarningIndicatorOffCriteria.	
Template Description	
This attribute defines the number of healing cycles for the WarningIndicatorOffCriteria	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticIndicator::DiagnosticIndicator.healingCycleCounterThreshold	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemEventParameter/DemIndicatorAttribute
BSW Parameter	**BSW Type**
DemIndicatorRef	EcucReferenceDef

BSW Description	
Reference to the used indicator.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemEventParameter
BSW Parameter	**BSW Type**
DemOBDGroupingAssociativeEventsRef	EcucReferenceDef
BSW Description	
This parameter defines a reference which points to a representative event of one group of associate events. The "reverence event" must refer to it self. Note: One event is only allowed to be reverenced to only one group of associate events.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemEventParameter
BSW Parameter	**BSW Type**
DemOperationCycleRef	EcucReferenceDef
BSW Description	
Kind of operation cycle for the event (e.g. power cycle, driving cycle, ...)	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemEventParameter
BSW Parameter	**BSW Type**
DemReportBehavior	EcucEnumerationParamDef
BSW Description	

Indicates the reporting behavior of the BSW Module (DemEventKind == DEM_EVENT_KIND_BSW) in order to determine the size of the reporting queue. If the parameter is not defined it means REPORT_BEFORE_INIT.
Template Description
This switch indicates whether or not the BSW module is allowed to report the related Events before Dem_Init().
M2 Parameter
CommonStructure::ServiceNeeds::DiagnosticEventNeeds.reportBehavior

Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	up_Dem_00003

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemEventParameter/DemReportBehavior

BSW Parameter	**BSW Type**
REPORT_AFTER_INIT	EcucEnumerationLiteralDef
BSW Description	
Indicates that the Event will not be reported before Dem_Init().	
Template Description	
This allows reporting related events after initialization	
M2 Parameter	
CommonStructure::ServiceNeeds::ReportBehaviorEnum.reportAfterInit	

Mapping Rule	**Mapping Type**
	full
Mapping Status	**Mapping ID**
valid	up_Dem_00005

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemEventParameter/DemReportBehavior

BSW Parameter	**BSW Type**
REPORT_BEFORE_INIT	EcucEnumerationLiteralDef
BSW Description	
Indicates that the Event may be reported before Dem_Init().	
Template Description	
This allows reporting related events before initialization	
M2 Parameter	
CommonStructure::ServiceNeeds::ReportBehaviorEnum.reportBeforeInit	

Mapping Rule	**Mapping Type**
	full
Mapping Status	**Mapping ID**
valid	up_Dem_00004

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemEventParameter

BSW Parameter	**BSW Type**
DemStorageConditionGroupRef	EcucReferenceDef
BSW Description	
References a storage condition group.	

Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context	
Dem	Dem/DemConfigSet	
BSW Parameter		**BSW Type**
DemJ1939Node		EcucParamConfContainerDef
BSW Description		
Contains the parameters for the support of a logical J1939 node.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemConfigSet/DemJ1939Node	
BSW Parameter		**BSW Type**
DemJ1939NmNodeRef		EcucSymbolicNameReferenceDef
BSW Description		
Reference to the corresponding J1939Nm node.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemConfigSet	
BSW Parameter		**BSW Type**
DemObdDTC		EcucParamConfContainerDef
BSW Description		
This container contains the configuration (parameters) for DemObdDTC.		
Template Description		
Unique Diagnostic Trouble Code value for OBD.		
M2 Parameter		
DiagnosticExtract::Dem::DiagnosticTroubleCode::DiagnosticTroubleCodeObd.obdDTCValue		
Mapping Rule		**Mapping Type**

1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemObdDTC
BSW Parameter	**BSW Type**
DemConsiderPtoStatus	EcucBooleanParamDef
BSW Description	
This parameter is TRUE, when the event is affected by the Dem PTO handling.	
Template Description	
This attribute describes the affection of the event by the Dem PTO handling. True: the event is affected by the Dem PTO handling. False: the event is not affected by the Dem PTO handling.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticTroubleCode::DiagnosticTroubleCodeObd.considerPtoStatus	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemObdDTC
BSW Parameter	**BSW Type**
DemDtcValue	EcucIntegerParamDef
BSW Description	
Unique Diagnostic Trouble Code value for OBD	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemConfigSet/DemObdDTC
BSW Parameter	**BSW Type**
DemEventOBDReadinessGroup	EcucEnumerationParamDef
BSW Description	
This parameter specifies the Event OBD Readiness group for PID $01 and PID $41 computation. This parameter is only applicable for emission-related ECUs.	
Template Description	
This attribute specifies the Event OBD Readiness group for PID $01 and PID $41 computation. This attribute is only applicable for emission-related ECUs.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticTroubleCode::DiagnosticTroubleCodeObd.eventObdReadiness Group	

Mapping Rule	Mapping Type
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemObdDTC
BSW Parameter	**BSW Type**
DemJ1939DTCValue	EcucIntegerParamDef
BSW Description	
Unique Diagnostic Trouble Code value for J1939 (consiting of SPN and FMI)	
Template Description	
Unique Diagnostic Trouble Code value for J1939 (consiting of SPN and FMI).	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticTroubleCode::DiagnosticTroubleCodeJ1939.j1939DtcValue	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet
BSW Parameter	**BSW Type**
DemPidClass	EcucParamConfContainerDef
BSW Description	
This container contains the different PIDs for the single global OBD relevant freeze frame class. It is assembled out of one or several data elements.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemConfigSet/DemPidClass
BSW Parameter	**BSW Type**
DemPidDataElement	EcucParamConfContainerDef
BSW Description	
This container contains the different data elements contained in the specific PID.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context	
Dem	Dem/DemConfigSet/DemPidClass/DemPidDataElement	
BSW Parameter		**BSW Type**
DemPidDataElementClassRef		EcucReferenceDef
BSW Description		
This reference contains the link to a data element class.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemConfigSet/DemPidClass	
BSW Parameter		**BSW Type**
DemPidIdentifier		EcucIntegerParamDef
BSW Description		
identifier of the PID		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem	
BSW Parameter		**BSW Type**
DemGeneral		EcucParamConfContainerDef
BSW Description		
This container contains the configuration (parameters) of the BSW Dem		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemGeneral	
BSW Parameter		**BSW Type**
DemAgingCycleCounterProcessing		EcucEnumerationParamDef
BSW Description		

This configuration switch defines, whether the aging counter is calculated Dem-internally or provided via Dem_SetAgingCycleCounterValue.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral
BSW Parameter	**BSW Type**
DemAgingRequieresTestedCycle	EcucBooleanParamDef
BSW Description	
Defines if the aging cycle counter is processed every aging cycles or if only tested aging cycle are considered. true: only tested aging cycle are considered for aging cycle counter false: aging cycle counter is processed every aging cycle	
Template Description	
Defines whether the aging cycle counter is processed every aging cycles or else only tested aging cycle are considered. If the attribute is set to TRUE: only tested aging cycle are considered for aging cycle counter. If the attribute is set to FALSE: aging cycle counter is processed every aging cycle.	
M2 Parameter	
DiagnosticExtract::DiagnosticCommonProps::DiagnosticCommonProps.agingRequiresTestedCycle	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral
BSW Parameter	**BSW Type**
DemAvailabilitySupport	EcucEnumerationParamDef
BSW Description	
This configuration switch defines, whether support for availability is enabled or not.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context	
Dem	Dem/DemGeneral	
BSW Parameter		**BSW Type**
DemBswErrorBufferSize		EcucIntegerParamDef
BSW Description		
Maximum number of elements in buffer for handling of BSW errors (ref. to SWS_Dem_00207).		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemGeneral	
BSW Parameter		**BSW Type**
DemCallbackDTCStatusChanged		EcucParamConfContainerDef
BSW Description		
The presence of this container indicates, that the Dem has access to a "DTCStatusChanged" callback, which the Dem will call to notify other components about the change in the status of a DTC. In case there is a DemCallbackDTCStatusChangedFnc, this parameter defines the name of the function that the Dem will call. In case there is no DemCallbackDTCStatusChangedFnc, the Dem will have an R-Port requiring the interface CallbackDTCStatusChanged whose name is generated by using the unique callback-prefix followed by the event name. Status change notifications are supported for DTCs in primary memory only.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemGeneral/DemCallbackDTCStatusChanged	
BSW Parameter		**BSW Type**
DemCallbackDTCStatusChangedFnc		EcucFunctionNameDef
BSW Description		
Function name of prototype "DTCStatusChanged". Note: If the parameter DemTriggerDcmReports is enabled, this parameter shall not be "Dcm_DemTriggerOnDTCStatus".		
Template Description		

M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral
BSW Parameter	**BSW Type**
DemClearDTCBehavior	EcucEnumerationParamDef
BSW Description	
Defines the clearing process of diagnostic information for volatile and non-volatile memory and the positive response handling for the Dcm module.	
Template Description	
This attribute defines the resulting UDS status byte for the related event, which shall not be cleared according to the ClearEventAllowed callback.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticEvent::DiagnosticEvent.clearEventBehavior	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral
BSW Parameter	**BSW Type**
DemClearDTCLimitation	EcucEnumerationParamDef
BSW Description	
Defines the supported Dem_<...>ClearDTC API scope.	
Template Description	
Defines the scope of the DEM_ClearDTC Api.	
M2 Parameter	
DiagnosticExtract::DiagnosticCommonProps::DiagnosticCommonProps.clearDtcLimitation	
Mapping Rule	**Mapping Type**
	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemClearDTCLimitation
BSW Parameter	**BSW Type**
DEM_ALL_SUPPORTED_DTCS	EcucEnumerationLiteralDef
BSW Description	
Dem_<...>ClearDTC accepts all supported DTC values, as well as all DTC values which are configured in DemGroupDTCs and DEM_DTC_GROUP_ALL_DTCS.	
Template Description	
DEM_ClearDtc API accepts all supported DTC values.	
M2 Parameter	
DiagnosticExtract::DiagnosticCommonProps::DiagnosticClearDtcLimitationEnum.allSupportedDtcs	

Mapping Rule	Mapping Type
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemClearDTCLimitation
BSW Parameter	**BSW Type**
DEM_ONLY_CLEAR_ALL_DTCS	EcucEnumerationLiteralDef
BSW Description	
Dem_<...>ClearDTC accepts ClearAllDTCs only.	
Template Description	
DEM_ClearDtc API accepts ClearAllDTCs only.	
M2 Parameter	
DiagnosticExtract::DiagnosticCommonProps::DiagnosticClearDtcLimitationEnum.clearAllDtcs	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral
BSW Parameter	**BSW Type**
DemDataElementClass	EcucChoiceContainerDef
BSW Description	
This container contains the configuration (parameters) for an internal/external data element class.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemDataElementClass
BSW Parameter	**BSW Type**
DemExternalCSDataElementClass	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) for an external client/server based data element class. It defines, how the Dem can obtain the value of the data element from either a SW-C or another BSW module. Whether a client/server port or a C function-call is used, is defined by DemDataElementUsePort.	
Template Description	
M2 Parameter	

Mapping Rule	Mapping Type
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemDataElementClass/DemExternalCSDataElementClass
BSW Parameter	**BSW Type**
DemDataElementDataSize	EcucIntegerParamDef
BSW Description	
Defines the size of the data element in bytes.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemDataElementClass/DemExternalCSDataElementClass
BSW Parameter	**BSW Type**
DemDataElementReadFnc	EcucFunctionNameDef
BSW Description	
In case of DemDataElementUsePort is false, this parameter defines the prototype of the C function "ReadDataElement" used to get the according value.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemDataElementClass/DemExternalCSDataElementClass
BSW Parameter	**BSW Type**
DemDataElementUsePort	EcucBooleanParamDef
BSW Description	
If the parameter is set to True, a R-Port is generated, to obtain the data element (interface DataServices_{Data}). If the parameter is set to False, the information is obtained by C function-call on another BSW module specified by the parameter DemDataElementReadFnc.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**

Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemDataElementClass
BSW Parameter	**BSW Type**
DemExternalSRDataElementClass	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) for an external sender/receiver based data element class. It defines, how the Dem can obtain the value of the data element from a SW-C, by using a sender/receiver port.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemDataElementClass/DemExternalSRDataElementClass
BSW Parameter	**BSW Type**
DemDataElementDataSize	EcucIntegerParamDef
BSW Description	
Defines the size of the data element in bits.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemDataElementClass/DemExternalSRDataElementClass
BSW Parameter	**BSW Type**
DemDataElementDataType	EcucEnumerationParamDef
BSW Description	
Provide the implementation data type of data belonging to a external data.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemDataElementClass/DemExternalSRDataElementClass
BSW Parameter	**BSW Type**
DemDataElementEndianness	EcucEnumerationParamDef
BSW Description	
Defines the endianness of the data belonging to an external data. If no DemDataElementEndianness is defined the value of DemDataElementDefaultEndianness is applicable.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemDataElementClass/DemExternalSRDataElementClass
BSW Parameter	**BSW Type**
DemDiagnosisScaling	EcucChoiceContainerDef
BSW Description	
This container contains the configuration (parameters) of an alternative Diagnosis Representation. Out if this the scaling between Diagnosis and ECU internal representation and vice versa can be calculated.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemDataElementClass/DemExternalSRDataElementClass/DemDiagnosisScaling
BSW Parameter	**BSW Type**
DemAlternativeDataInterface	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) of an alternative Diagnosis Representation by the means of a VariableDataPrototype in a DataInterface. Additionally a reference to PortInterfaceMapping can be defined which provide already the mapping rules between the VariableDataPrototype in a DataInterface used by the software component (DemSRDataElementClass) and the intended Diagnosis Representation defined by DemExternalSRDataElementClass.	
Template Description	
M2 Parameter	

Mapping Rule	Mapping Type
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemDataElementClass/DemExternalSRDataElementClass/DemDiagnosisScaling/DemAlternativeDataInterface
BSW Parameter	**BSW Type**
DemDataElement	EcucForeignReferenceDef
BSW Description	
Alternative Diagnosis Representation for the data defined by the means of a VariableDataPrototype in a DataInterface.	
Template Description	
M2 Parameter	

Mapping Rule	Mapping Type
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemDataElementClass/DemExternalSRDataElementClass/DemDiagnosisScaling/DemAlternativeDataInterface
BSW Parameter	**BSW Type**
DemPortInterfaceMapping	EcucForeignReferenceDef
BSW Description	
Optional reference to PortInterfaceMapping which defines the mapping rules.	
Template Description	
M2 Parameter	

Mapping Rule	Mapping Type
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemDataElementClass/DemExternalSRDataElementClass/DemDiagnosisScaling
BSW Parameter	**BSW Type**
DemAlternativeDataProps	EcucParamConfContainerDef
BSW Description	

This container contains the configuration (parameters) of an alternative Diagnosis Representation by the means of ECU configuration parameters. The physical unit of the alternative data representation is defined by the DataPrototype referenced by DemSRDataElementClass. Additionally the definition of a text table mapping can be a defined for DemDataTypeCategory TEXTTABLE and SCALE_LINEAR_AND_TEXTTABLE
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemDataElementClass/DemExternalSRDataElementClass/DemDiagnosisScaling/DemAlternativeDataProps

BSW Parameter	**BSW Type**
DemDataTypeCategory	EcucEnumerationParamDef
BSW Description	
Data category of the alternative Diagnosis Representation.	
Template Description	
M2 Parameter	

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemDataElementClass/DemExternalSRDataElementClass/DemDiagnosisScaling/DemAlternativeDataProps

BSW Parameter	**BSW Type**
DemLinearScale	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) of an linear scale of the alternative Diagnosis Representation.	
Template Description	
M2 Parameter	

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**

Dem	Dem/DemGeneral/DemDataElementClass/DemExternalSRDataElementClass/DemDiagnosisScaling/DemAlternativeDataProps/DemLinearScale

BSW Parameter	BSW Type
DemDiagnosisRepresentationDataLowerRange	EcucFloatParamDef

BSW Description
Lower Range for this scale of the data in the alternative Diagnosis Representation.

Template Description

M2 Parameter

Mapping Rule	Mapping Type
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemDataElementClass/DemExternalSRDataElementClass/DemDiagnosisScaling/DemAlternativeDataProps/DemLinearScale

BSW Parameter	BSW Type
DemDiagnosisRepresentationDataOffset	EcucFloatParamDef

BSW Description
Data offset of the alternative Diagnosis Representation for this scale.

Template Description

M2 Parameter

Mapping Rule	Mapping Type
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemDataElementClass/DemExternalSRDataElementClass/DemDiagnosisScaling/DemAlternativeDataProps/DemLinearScale

BSW Parameter	BSW Type
DemDiagnosisRepresentationDataResolution	EcucFloatParamDef

BSW Description
Data resolution of the alternative Diagnosis Representation for this scale.

Template Description

M2 Parameter

Mapping Rule	Mapping Type
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemDataElementClass/DemExternalSRDataElementClass/DemDiagnosisScaling/DemAlternativeDataProps/DemLinearScale

BSW Parameter	BSW Type
DemDiagnosisRepresentationDataUpperRange	EcucFloatParamDef
BSW Description	
Upper Range for this scale of the data in the alternative Diagnosis Representation.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemDataElementClass/DemExternalSRDataElementClass/DemDiagnosisScaling/DemAlternativeDataProps
BSW Parameter	**BSW Type**
DemTextTableMapping	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) of the mapping a DataPrototype typed by AutosarDataType that refer to a CompuMethods of category TEXTTABLE or SCALE_LINEAR_AND_TEXTTABLE. Each DemTextTableMapping defines a value pair which is used to map the ECU internal value (DemInternalDataValue) to the vale used in the diagnosis representation (DemDiagnosisRepresentationDataValue) and vice versa. The set of all DemTextTableMapping defines the whole mapping of an data.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemDataElementClass/DemExternalSRDataElementClass/DemDiagnosisScaling/DemAlternativeDataProps/DemTextTableMapping
BSW Parameter	**BSW Type**
DemDiagnosisRepresentationDataValue	EcucIntegerParamDef
BSW Description	
The data value in the diagnosis representation.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemDataElementClass/DemExternalSRDataElementClass/DemDiagnosisScaling/DemAlternativeDataProps/DemTextTableMapping
BSW Parameter	**BSW Type**
DemInternalDataValue	EcucIntegerParamDef
BSW Description	
The ECU internal data value.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemDataElementClass/DemExternalSRDataElementClass/DemDiagnosisScaling
BSW Parameter	**BSW Type**
DemAlternativeDataType	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) of an alternative Diagnosis Representation by the means of an ApplicationDataType. Additionally the definition of a text table mapping can be a defined for ApplicationDataTypes of category TEXTTABLE and SCALE_LINEAR_AND_TEXTTABLE.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemDataElementClass/DemExternalSRDataElementClass/DemDiagnosisScaling/DemAlternativeDataType
BSW Parameter	**BSW Type**
DemApplicationDataType	EcucForeignReferenceDef
BSW Description	
Alternative Diagnosis Representation for the data defined by the means of a ApplicationPrimitive-DataType of category VALUE or BOOLEAN.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemDataElementClass/DemExternalSRDataElementClass/DemDiagnosisScaling/DemAlternativeDataType

BSW Parameter	BSW Type
DemTextTableMapping	EcucParamConfContainerDef

BSW Description

This container contains the configuration (parameters) of the mapping a DataPrototype typed by AutosarDataType that refer to a CompuMethods of category TEXTTABLE or SCALE_LINEAR_AND_TEXTTABLE.

Each DemTextTableMapping defines a value pair which is used to map the ECU internal value (DemInternalDataValue) to the vale used in the diagnosis representation (DemDiagnosisRepresentationDataValue) and vice versa.

The set of all DemTextTableMapping defines the whole mapping of an data.

Template Description

M2 Parameter

Mapping Rule	Mapping Type
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemDataElementClass/DemExternalSRDataElementClass/DemDiagnosisScaling/DemAlternativeDataType/DemTextTableMapping

BSW Parameter	BSW Type
DemDiagnosisRepresentationDataValue	EcucIntegerParamDef

BSW Description

The data value in the diagnosis representation.

Template Description

M2 Parameter

Mapping Rule	Mapping Type
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemDataElementClass/DemExternalSRDataElementClass/DemDiagnosisScaling/DemAlternativeDataType/DemTextTableMapping

BSW Parameter	BSW Type
DemInternalDataValue	EcucIntegerParamDef

BSW Description

The ECU internal data value.

Template Description

M2 Parameter

Mapping Rule	Mapping Type

Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemDataElementClass/DemExternalSRDataElementClass
BSW Parameter	**BSW Type**
DemSRDataElementClass	EcucChoiceContainerDef
BSW Description	
This container defines the source of data in a provided port which shall be read for a external data element This container shall contain either one DemSubElementInDataElementInstance OR DemDataElementInstance OR DemSubElementInImplDataElementInstance reference.	
Template Description	
M2 Parameter	

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemDataElementClass/DemExternalSRDataElementClass/ DemSRDataElementClass
BSW Parameter	**BSW Type**
DemDataElementInstance	EcucParamConfContainerDef
BSW Description	
Instance Reference to the primitive data in a port where the data element is typed with an ApplicationPrimitveDataType or an ImplementationDataType.	
Template Description	
M2 Parameter	

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemDataElementClass/DemExternalSRDataElementClass/ DemSRDataElementClass/DemDataElementInstance
BSW Parameter	**BSW Type**
DemDataElementInstanceRef	EcucInstanceReferenceDef
BSW Description	

Instance Reference to the primitive data which shall be read or written. Supported are VariableDataPrototypes in SenderReceiverInterfaces and NvDataInterfaces and ParameterDataPrototypes in ParameterInterfaces (read only). This reference is applicable if the AutosarDataPrototype is typed with a ApplicationPrimitiveDataType of category VALUE or BOOLEAN or if the AutosarDataPrototype is typed with a ImplementationDataType of category VALUE or TYPE_REFERENCE that in turn boils down to VALUE
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemDataElementClass/DemExternalSRDataElementClass/DemSRDataElementClass

BSW Parameter	**BSW Type**
DemSubElementInDataElementInstance	EcucParamConfContainerDef

BSW Description
Instance Reference to the primitve sub-element (at any level) of composite data in a port where the data element is typed with an ApplicationCompositeDataType.
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemDataElementClass/DemExternalSRDataElementClass/DemSRDataElementClass/DemSubElementInDataElementInstance

BSW Parameter	**BSW Type**
DemSubElementInDataElementInstanceRef	EcucInstanceReferenceDef

BSW Description
Instance Reference to the primitve sub-element (at any level) of composite data in a port which shall be read or written. Supported are VariableDataPrototypes in SenderReceiverInterfaces and NvDataInterfaces and ParameterDataPrototypes in ParameterInterfaces (read only). This reference is applicable if the AutosarDataPrototype is typed with a ApplicationCompositeDataType.
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemDataElementClass/DemExternalSRDataElementClass/DemSRDataElementClass
BSW Parameter	**BSW Type**
DemSubElementInImplDataElementInstance	EcucParamConfContainerDef
BSW Description	
Instance Reference to the primitve sub-element (at any level) of composite data in a port where the data element is typed with an ImplementationDataType of category STRUCTURE or ARRAY.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemDataElementClass/DemExternalSRDataElementClass/DemSRDataElementClass/DemSubElementInImplDataElementInstance
BSW Parameter	**BSW Type**
DemSubElementInImplDataElementInstanceRef	EcucInstanceReferenceDef
BSW Description	
Instance Reference to the primitve sub-element (at any level) of composite data in a port which shall be read or written. Supported are VariableDataPrototypes in SenderReceiverInterfaces and NvDataInterfaces and ParameterDataPrototypes in ParameterInterfaces (read only). This reference is applicable if the AutosarDataPrototype is typed with a ImplementationDataType of category STRUCTURE or ARRAY. Please note that in case of ARRAY the index attribute in the target reference has to be set to select a single array element.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemDataElementClass
BSW Parameter	**BSW Type**
DemInternalDataElementClass	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) for an internal data element class.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemDataElementClass/DemInternalDataElementClass
BSW Parameter	**BSW Type**
DemDataElementDataSize	EcucIntegerParamDef
BSW Description	
Defines the size of the data element in bytes.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemDataElementClass/DemInternalDataElementClass
BSW Parameter	**BSW Type**
DemInternalDataElement	EcucEnumerationParamDef
BSW Description	
This parameter defines the Dem-internal data value, which is mapped to the data element.	
Template Description	
This represents the ability to further specify the access within the Dem.	
M2 Parameter	
DiagnosticExtract::ServiceMapping::DiagnosticDemProvidedDataMapping.dataProvider	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral
BSW Parameter	**BSW Type**
DemDataElementDefaultEndianness	EcucEnumerationParamDef
BSW Description	
Defines the default endianness of the data belonging to a data element which is applicable if the DemExternalSRDataElementClass does not define a endianness.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral
BSW Parameter	**BSW Type**
DemDebounceCounterBasedSupport	EcucBooleanParamDef
BSW Description	
This configuration switch defines, whether support for counter based debouncing is enabled or not. true: counter based debouncing support is enabled false: counter based debouncing support is disabled	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral
BSW Parameter	**BSW Type**
DemDebounceTimeBasedSupport	EcucBooleanParamDef
BSW Description	
This configuration switch defines, whether support for time based debouncing is enabled or not. true: time based debouncing support is enabled false: time based debouncing support is disabled	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral
BSW Parameter	**BSW Type**
DemDevErrorDetect	EcucBooleanParamDef
BSW Description	
Switches the Default Error Tracer (Det) detection and notification ON or OFF. * true: enabled (ON). * false: disabled (OFF).	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context	
Dem	Dem/DemGeneral	
BSW Parameter		**BSW Type**
DemDidClass		EcucParamConfContainerDef
BSW Description		
This container contains the configuration (parameters) for a data Id class. It is assembled out of one or several data elements.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemGeneral/DemDidClass	
BSW Parameter		**BSW Type**
DemDidDataElementClassRef		EcucReferenceDef
BSW Description		
This reference contains the link to a data element class.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemGeneral/DemDidClass	
BSW Parameter		**BSW Type**
DemDidIdentifier		EcucIntegerParamDef
BSW Description		
Identifier of the Data ID.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemGeneral	
BSW Parameter		**BSW Type**
DemDtcStatusAvailabilityMask		EcucIntegerParamDef

BSW Description	
Mask for the supported DTC status bits by the Dem. This mask is used by UDS service 0x19.	
Template Description	
Mask for the supported DTC status bits by the Dem.	
M2 Parameter	
DiagnosticExtract::DiagnosticCommonProps::DiagnosticCommonProps.dtcStatusAvailabilityMask	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral
BSW Parameter	**BSW Type**
DemEnableCondition	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) for enable conditions.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemEnableCondition
BSW Parameter	**BSW Type**
DemEnableConditionId	EcucIntegerParamDef
BSW Description	
Defines a unique enable condition Id. This parameter should not be changeable by user, because the Id should be generated by Dem itself to prevent gaps and multiple use of an Id. The enable conditions should be sequentially ordered beginning with 0 and no gaps in between.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemEnableCondition
BSW Parameter	**BSW Type**
DemEnableConditionStatus	EcucBooleanParamDef
BSW Description	

Defines the initial status for enable or disable of acceptance of event reports of a diagnostic event. The value is the initialization after power up (before this condition is reported the first time). true: acceptance of a diagnostic event enabled false: acceptance of a diagnostic event disabled
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral

BSW Parameter	**BSW Type**
DemEnableConditionGroup	EcucParamConfContainerDef

BSW Description
This container contains the configuration (parameters) for enable condition groups.
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemEnableConditionGroup

BSW Parameter	**BSW Type**
DemEnableConditionRef	EcucReferenceDef

BSW Description
References an enable condition.
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral

BSW Parameter	**BSW Type**
DemEnvironmentDataCapture	EcucEnumerationParamDef

BSW Description
DemEnvironmentDataCapture defines the point in time, when the data actually is captured.

Template Description	
This attribute determines the point in time, when the data actually is captured.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticTroubleCode::DiagnosticTroubleCodeProps.environmentCaptureToReporting	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemEnvironmentDataCapture
BSW Parameter	**BSW Type**
DEM_CAPTURE_ASYNCHRONOUS_TO_REPORTING	EcucEnumerationLiteralDef
BSW Description	
The data capturing is postponed to the next cycle of the Dem_Mainfunction. (This means that there is a minimum delay between report of the failure and capturing the data).	
Template Description	
The data capturing is postponed to the next cycle of the Dem_Mainfunction. (This means that there is a minimum delay between report of the failure and capturing the data.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticTroubleCode::EnvironmentCaptureToReportingEnum.captureAsynchronousToReporting	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemEnvironmentDataCapture
BSW Parameter	**BSW Type**
DEM_CAPTURE_SYNCHRONOUS_TO_REPORTING	EcucEnumerationLiteralDef
BSW Description	
The data is captured immediately within the reporting function (i.e. in the context of the setEventStatus/reportErrorStatus function).	
Template Description	
The data is captured immediately within the reporting function (i.e. in the context of the setEventStatus/reportErrorStatus function).	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticTroubleCode::EnvironmentCaptureToReportingEnum.captureSynchronousToReporting	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral
BSW Parameter	**BSW Type**
DemEventCombinationSupport	EcucEnumerationParamDef

BSW Description	
This parameter defines the type of event combination supported by the Dem.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context	
Dem	Dem/DemGeneral	
BSW Parameter		**BSW Type**
DemEventDisplacementStrategy		EcucEnumerationParamDef
BSW Description		
This configuration switch defines, whether support for event displacement is enabled or not, and which displacement strategy is followed.		
Template Description		
This attribute defines, whether support for event displacement is enabled or not, and which displacement strategy is followed.		
M2 Parameter		
DiagnosticExtract::DiagnosticCommonProps::DiagnosticCommonProps.eventDisplacementStrategy		
Mapping Rule		**Mapping Type**
1:1 mapping		full
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemGeneral/DemEventDisplacementStrategy	
BSW Parameter		**BSW Type**
DEM_DISPLACEMENT_FULL		EcucEnumerationLiteralDef
BSW Description		
Event memory entry displacement is enabled, by consideration of priority active/passive status, and occurrence.		
Template Description		
Event memory entry displacement is enabled, by consideration of priority active/passive status, and occurrence.		
M2 Parameter		
DiagnosticExtract::DiagnosticCommonProps::DiagnosticEventDisplacementStrategyEnum.full		
Mapping Rule		**Mapping Type**
1:1 mapping		full
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemGeneral/DemEventDisplacementStrategy	
BSW Parameter		**BSW Type**
DEM_DISPLACEMENT_NONE		EcucEnumerationLiteralDef

BSW Description	
Event memory entry displacement is disabled.	
Template Description	
Event memory entry displacement is disabled.	
M2 Parameter	
DiagnosticExtract::DiagnosticCommonProps::DiagnosticEventDisplacementStrategyEnum.none	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemEventDisplacementStrategy
BSW Parameter	**BSW Type**
DEM_DISPLACEMENT_PRIO_OCC	EcucEnumerationLiteralDef
BSW Description	
Event memory entry displacement is enabled, by consideration of priority and occurrence (but without active/passive status).	
Template Description	
Event memory entry displacement is enabled, by consideration of priority and occurrence (but without active/passive status).	
M2 Parameter	
DiagnosticExtract::DiagnosticCommonProps::DiagnosticEventDisplacementStrategyEnum.prioOcc	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral
BSW Parameter	**BSW Type**
DemEventMemoryEntryStorageTrigger	EcucEnumerationParamDef
BSW Description	
Configures the primary trigger to allocate an event memory entry.	
Template Description	
Describes the primary trigger to allocate an event memory entry.	
M2 Parameter	
DiagnosticExtract::DiagnosticCommonProps::DiagnosticCommonProps.memoryEntryStorageTrigger	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral
BSW Parameter	**BSW Type**
DemExtendedDataClass	EcucParamConfContainerDef
BSW Description	
This class contains the combinations of extended data records for an extended data class.	

Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemExtendedDataClass
BSW Parameter	**BSW Type**
DemExtendedDataRecordClassRef	EcucReferenceDef
BSW Description	
This reference contains the link to an extended data class record.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral
BSW Parameter	**BSW Type**
DemExtendedDataRecordClass	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) for an extended data record class. It is assembled out of one or several data elements.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemExtendedDataRecordClass
BSW Parameter	**BSW Type**
DemDataElementClassRef	EcucReferenceDef
BSW Description	
This reference contains the link to a data element class.	
Template Description	
M2 Parameter	

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemExtendedDataRecordClass
BSW Parameter	**BSW Type**
DemExtendedDataRecordNumber	EcucIntegerParamDef
BSW Description	
This configuration parameter specifies an unique identifier for an extended data record. One or more extended data records can be assigned to one diagnostic event/DTC. 0x00 is reserved by ISO (therefore the minimal value equals 1) 0xF0 to 0xFF are reserved by ISO (therefore the maximal value equals 239)	
Template Description	
This attribute specifies an unique identifier for an extended data record.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticExtendedDataRecord::DiagnosticExtendedDataRecord.record Number	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemExtendedDataRecordClass
BSW Parameter	**BSW Type**
DemExtendedDataRecordTrigger	EcucEnumerationParamDef
BSW Description	
Defines the trigger to store the ExtendedDataRecord.	
Template Description	
This attribute specifies the primary trigger to allocate an event memory entry.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticExtendedDataRecord::DiagnosticExtendedDataRecord.trigger	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemExtendedDataRecordClass/DemExtendedDataRecord Trigger
BSW Parameter	**BSW Type**
DEM_TRIGGER_ON_CONFIRMED	EcucEnumerationLiteralDef
BSW Description	
ExtendedDataRecord will be stored when the event status confirmed bit changes from 0 to 1.	

Template Description	
capture on "Confirmed"	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticFreezeFrame::DiagnosticRecordTriggerEnum.confirmed	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemExtendedDataRecordClass/DemExtendedDataRecordTrigger
BSW Parameter	**BSW Type**
DEM_TRIGGER_ON_FDC_THRESHOLD	EcucEnumerationLiteralDef
BSW Description	
ExtendedDataRecord will be stored when the FDC reaches its threshold.	
Template Description	
capture on "FDC Threshold"	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticFreezeFrame::DiagnosticRecordTriggerEnum.fdcThreshold	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemExtendedDataRecordClass/DemExtendedDataRecordTrigger
BSW Parameter	**BSW Type**
DEM_TRIGGER_ON_PENDING	EcucEnumerationLiteralDef
BSW Description	
ExtendedDataRecord will be stored when the event status pending bit changes from 0 to 1.	
Template Description	
capture on "Pending"	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticFreezeFrame::DiagnosticRecordTriggerEnum.pending	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemExtendedDataRecordClass/DemExtendedDataRecordTrigger
BSW Parameter	**BSW Type**
DEM_TRIGGER_ON_TEST_FAILED	EcucEnumerationLiteralDef
BSW Description	
ExtendedDataRecord will be stored when the event status test failed bit changes from 0 to 1.	
Template Description	
capture on "Test Failed"	

M2 Parameter	
DiagnosticExtract::Dem::DiagnosticFreezeFrame::DiagnosticRecordTriggerEnum.testFailed	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemExtendedDataRecordClass
BSW Parameter	**BSW Type**
DemExtendedDataRecordUpdate	EcucEnumerationParamDef
BSW Description	
This extended data record is captured if the configured trigger condition in "DemExtended-DataRecordTrigger" is fulfilled.	
Template Description	
This attribute defines when an extended data record is captured. True: This extended data record is captured every time. False: This extended data record is only captured for new event memory entries.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticExtendedDataRecord::DiagnosticExtendedDataRecord.update	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral
BSW Parameter	**BSW Type**
DemFreezeFrameClass	EcucParamConfContainerDef
BSW Description	
This container contains the combinations of DIDs for a non OBD2 and WWH-OBD relevant freeze frame class.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemFreezeFrameClass
BSW Parameter	**BSW Type**
DemDidClassRef	EcucReferenceDef
BSW Description	
Reference to the DID elements which shall be contained in the freeze frame.	
Template Description	
M2 Parameter	

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral
BSW Parameter	**BSW Type**
DemFreezeFrameRecNumClass	EcucParamConfContainerDef
BSW Description	
This container contains a list of dedicated, different freeze frame record numbers assigned to an event. The order of record numbers in this list is assigned to the chronological order of the according freeze frame records. dependency: DemTypeOfFreezeFrameRecordNumeration = DEM_FF_RECNUM_CONFIGURED	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemFreezeFrameRecNumClass
BSW Parameter	**BSW Type**
DemFreezeFrameRecordClassRef	EcucReferenceDef
BSW Description	
This parameter references record number(s) for a freeze frame record.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral
BSW Parameter	**BSW Type**
DemFreezeFrameRecordClass	EcucParamConfContainerDef
BSW Description	
This container contains a list of dedicated, different freeze frame record numbers.	
Template Description	
M2 Parameter	

Mapping Rule	Mapping Type
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemFreezeFrameRecordClass
BSW Parameter	**BSW Type**
DemFreezeFrameRecordNumber	EcucIntegerParamDef
BSW Description	
This parameter defines a record number for a freeze frame record. This record number is unique per freeze frame record number class.	
Template Description	
This attribute defines a record number for a freeze frame record.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticFreezeFrame::DiagnosticFreezeFrame.recordNumber	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemFreezeFrameRecordClass
BSW Parameter	**BSW Type**
DemFreezeFrameRecordTrigger	EcucEnumerationParamDef
BSW Description	
Defines the trigger to store the FreezeFrameRecordTrigger.	
Template Description	
This attribute defines the primary trigger to allocate an event memory entry.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticFreezeFrame::DiagnosticFreezeFrame.trigger	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemFreezeFrameRecordClass
BSW Parameter	**BSW Type**
DemFreezeFrameRecordUpdate	EcucEnumerationParamDef
BSW Description	
This parameter defines the case, when the freeze frame record is stored/updated.	
Template Description	
This attribute defines the approach when the freeze frame record is stored/updated. True: FreezeFrame record is captured every time. False: FreezeFrame record is only captured for new event memory entries.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticFreezeFrame::DiagnosticFreezeFrame.update	
Mapping Rule	**Mapping Type**
1:1 mapping	full

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral
BSW Parameter	**BSW Type**
DemGeneralInterfaceSupport	EcucBooleanParamDef
BSW Description	
The interfaces GeneralEvtInfo, GeneralCallbackEventDataChanged and GeneralCallbackEventStatusChange are provided if DemGeneralInterfaceSupport is equal to true.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral
BSW Parameter	**BSW Type**
DemGeneralJ1939	EcucParamConfContainerDef
BSW Description	
This container contains the general J1939-specific configuration (parameters) of the Dem module. If the container exists the J1939 support is enabled.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemGeneralJ1939
BSW Parameter	**BSW Type**
DemAmberWarningLampIndicatorRef	EcucReferenceDef
BSW Description	
This parameter defines the indicator representing the AmberWarningLamp . This parameter may be used for ECUs supporting J1939.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemGeneralJ1939
BSW Parameter	**BSW Type**
DemCallbackJ1939DTCStatusChanged	EcucParamConfContainerDef
BSW Description	
The presence of this container indicates, that the Dem has access to a "DTCStatusChanged" callback, which the Dem will call to notify other components about the change in the status of a DTC. In case there is a DemCallbackDTCStatusChangedFnc, this parameter defines the name of the function that the Dem will call. In case there is no DemCallbackDTCStatusChangedFnc, the Dem will have an R-Port requiring the interface CallbackDTCStatusChanged whose name is generated by using the unique callback-prefix followed by the event name. Status change notifications are supported for DTCs in primary memory only.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemGeneralJ1939/DemCallbackJ1939DTCStatusChanged
BSW Parameter	**BSW Type**
DemCallbackDTCStatusChangedFnc	EcucFunctionNameDef
BSW Description	
Function name of prototype "DTCStatusChanged". Note: If the parameter DemTriggerDcmReports is enabled, this parameter shall not be "Dcm_DemTriggerOnDTCStatus".	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemGeneralJ1939
BSW Parameter	**BSW Type**
DemJ1939ClearDtcSupport	EcucBooleanParamDef
BSW Description	

This configuration switch defines whether clearing J1939 DTCs (DM03 und DM11) is supported or not. This switches on and off the API Dem_J1939DcmClearDTC.
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemGeneralJ1939

BSW Parameter	**BSW Type**
DemJ1939Dm31Support	EcucBooleanParamDef

BSW Description
This configuration switch defines whether J1939 DM31 is supported or not. This switches on and off the APIs Dem_J1939DcmFirstDTCwithLampStatus and Dem_J1939DcmGetNextDTCwithLampStatus. .
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemGeneralJ1939

BSW Parameter	**BSW Type**
DemJ1939ExpandedFreezeFrameSupport	EcucBooleanParamDef

BSW Description
This configuration switch defines whether J1939 expanded freeze frames are supported or not. This switches on and off the APIs Dem_J1939DcmSetFreezeFrameFilter, Dem_J1939DcmGetNextFreezeFrame and Dem_J1939DcmGetNextSPNInFreezeFrame.
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**

Dem	Dem/DemGeneral/DemGeneralJ1939	
BSW Parameter		**BSW Type**
DemJ1939FreezeFrameClass		EcucParamConfContainerDef
BSW Description		
This container contains the combinations of SPNs s for a J1939 relevant freeze frame.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	**BSW Context**	
Dem	Dem/DemGeneral/DemGeneralJ1939/DemJ1939FreezeFrameClass	
BSW Parameter		**BSW Type**
DemSPNClassRef		EcucReferenceDef
BSW Description		
Reference to an SPN. This reference defines requiresIndex = true since it represents a ordered list of references where the order describes the order of single SPNs in the J1939 Freeze Frame.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	**BSW Context**	
Dem	Dem/DemGeneral/DemGeneralJ1939	
BSW Parameter		**BSW Type**
DemJ1939FreezeFrameSupport		EcucBooleanParamDef
BSW Description		
This configuration switch defines whether J1939 freeze frames are supported or not. This switches on and off the APIs Dem_J1939DcmSetFreezeFrameFilter and Dem_J1939DcmGetNextFreezeFrame.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemGeneralJ1939

BSW Parameter	BSW Type
DemJ1939RatioSupport	EcucBooleanParamDef
BSW Description	
This configuration switch defines whether J1939 performance ratios are supported or not. This switches on and off the APIs Dem_J1939DcmSetRatioFilter and Dem_J1939DcmGetNextFilteredRatio.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemGeneralJ1939
BSW Parameter	**BSW Type**
DemJ1939Readiness1Support	EcucBooleanParamDef
BSW Description	
This configuration switch defines whether J1939 diagnostic readiness 1 is supported or not. This switches on and off the API Dem_J1939DcmReadDiagnosticReadiness1.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemGeneralJ1939
BSW Parameter	**BSW Type**
DemJ1939Readiness2Support	EcucBooleanParamDef
BSW Description	
This configuration switch defines whether J1939 diagnostic readiness 2 is supported or not. This switches on and off the API Dem_J1939DcmReadDiagnosticReadiness2.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context	
Dem	Dem/DemGeneral/DemGeneralJ1939	
BSW Parameter		**BSW Type**
DemJ1939Readiness3Support		EcucBooleanParamDef
BSW Description		
This configuration switch defines whether J1939 diagnostic readiness 3 is supported or not. This switches on and off the API Dem_J1939DcmReadDiagnosticReadiness3.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemGeneral/DemGeneralJ1939	
BSW Parameter		**BSW Type**
DemJ1939ReadingDtcSupport		EcucBooleanParamDef
BSW Description		
This configuration switch defines whether J1939 DTC readout is supported or not. This switches on and off the APIs Dem_J1939DcmSetDTCFilter, Dem_J1939DcmGetNumberOfFilteredDTC and Dem_J1939DcmGetNextFilteredDTC.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemGeneral/DemGeneralJ1939	
BSW Parameter		**BSW Type**
DemProtectLampIndicatorRef		EcucReferenceDef
BSW Description		
This parameter defines the indicator representing the ProtectLamp. This parameter may be used for ECUs supporting J1939.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemGeneral/DemGeneralJ1939	
BSW Parameter		**BSW Type**
DemRedStopLampIndicatorRef		EcucReferenceDef
BSW Description		
This parameter defines the indicator representing the RedStopLamp. This parameter may be used for ECUs supporting J1939.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemGeneral/DemGeneralJ1939	
BSW Parameter		**BSW Type**
DemSPNClass		EcucParamConfContainerDef
BSW Description		
This container contains the configuration (parameters) for a SPN.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemGeneral/DemGeneralJ1939/DemSPNClass	
BSW Parameter		**BSW Type**
DemSPNDataElementClassRef		EcucReferenceDef
BSW Description		
This reference contains the link to a data element class.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemGeneral/DemGeneralJ1939/DemSPNClass	
BSW Parameter		**BSW Type**
DemSPNId		EcucIntegerParamDef

BSW Description	
Suspect parameter number	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context	
Dem	Dem/DemGeneral	
BSW Parameter		**BSW Type**
DemGeneralOBD		EcucParamConfContainerDef
BSW Description		
This container contains the general OBD-specific configuration (parameters) of the Dem module.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemGeneral/DemGeneralOBD	
BSW Parameter		**BSW Type**
DemCallbackOBDDTCStatusChanged		EcucParamConfContainerDef
BSW Description		
The presence of this container indicates, that the Dem has access to a "DTCStatusChanged" callback, which the Dem will call to notify other components about the change in the status of a DTC. In case there is a DemCallbackDTCStatusChangedFnc, this parameter defines the name of the function that the Dem will call. In case there is no DemCallbackDTCStatusChangedFnc, the Dem will have an R-Port requiring the interface CallbackDTCStatusChanged whose name is generated by using the unique callback-prefix followed by the event name. Status change notifications are supported for DTCs in primary memory only.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemGeneralOBD/DemCallbackOBDDTCStatusChanged
BSW Parameter	**BSW Type**
DemCallbackDTCStatusChangedFnc	EcucFunctionNameDef
BSW Description	
Function name of prototype "DTCStatusChanged". Note: If the parameter DemTriggerDcmReports is enabled, this parameter shall not be "Dcm_DemTriggerOnDTCStatus".	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemGeneralOBD
BSW Parameter	**BSW Type**
DemOBDCentralizedPID21Handling	EcucBooleanParamDef
BSW Description	
Switch to enable the centralized handling of PID $21. true: centralized handling of PID $21 enabled false: centralized handling of PID $21 disabled	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemGeneralOBD
BSW Parameter	**BSW Type**
DemOBDCentralizedPID31Handling	EcucBooleanParamDef
BSW Description	
Switch to enable the centralized handling of PID $31. true: centralized handling of PID $31 enabled false: centralized handling of PID $31 disabled	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**

Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemGeneralOBD
BSW Parameter	**BSW Type**
DemOBDCompliancy	EcucIntegerParamDef
BSW Description	
Configuration value to define the appropriate value to PID$1C "OBD requirements to which vehicle or engine is certified." according to the respective standards, e.g. OBD, OBDII, JOBD etc. Notice as well J1979 or the "DiagnosticReadiness 1" DM05 message of J1939-73	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemGeneralOBD
BSW Parameter	**BSW Type**
DemOBDDestinationOfEventsRef	EcucChoiceReferenceDef
BSW Description	
The destination of events assigns where the OBD events shall be stored.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemGeneralOBD
BSW Parameter	**BSW Type**
DemOBDEngineType	EcucEnumerationParamDef
BSW Description	
Switch to provide either Gasoline or Diesel parameters.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemGeneralOBD
BSW Parameter	**BSW Type**
DemOBDEventDisplacement	EcucBooleanParamDef
BSW Description	
Activate/Deactivate a different displacement behavior for OBD events. OBD events with special Conditions (e.g. Pending, MIL_On...) shall not be displaced.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemGeneralOBD
BSW Parameter	**BSW Type**
DemOBDInputAcceleratorPedalInformation	EcucChoiceReferenceDef
BSW Description	
Input variable for the accelerator padal information, which is assigned to a specific data element used as interface for the Dem-internal PID calculations.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemGeneralOBD
BSW Parameter	**BSW Type**
DemOBDInputAmbientPressure	EcucChoiceReferenceDef
BSW Description	
Input variable for the ambient pressure, which is assigned to a specific data element used as interface for the Dem-internal PID calculations.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context

Dem	Dem/DemGeneral/DemGeneralOBD	
BSW Parameter		**BSW Type**
DemOBDInputAmbientTemperature		EcucChoiceReferenceDef
BSW Description		
Input variable for the ambient temperature, which is assigned to a specific data element used as interface for the Dem-internal PID calculations.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	**BSW Context**	
Dem	Dem/DemGeneral/DemGeneralOBD	
BSW Parameter		**BSW Type**
DemOBDInputDistanceInformation		EcucChoiceReferenceDef
BSW Description		
Input variable for the distance information, which is assigned to a specific data element used as interface for the Dem-internal PID calculations.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	**BSW Context**	
Dem	Dem/DemGeneral/DemGeneralOBD	
BSW Parameter		**BSW Type**
DemOBDInputEngineSpeed		EcucChoiceReferenceDef
BSW Description		
Input variable for the engine speed, which is assigned to a specific data element used as interface for the Dem-internal PID calculations.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	**BSW Context**	
Dem	Dem/DemGeneral/DemGeneralOBD	
BSW Parameter		**BSW Type**

DemOBDInputEngineTemperature	EcucChoiceReferenceDef
BSW Description	
Input variable for the engine temperature, which is assigned to a specific data element used as interface for the Dem-internal PID calculations.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemGeneralOBD
BSW Parameter	**BSW Type**
DemOBDInputProgrammingEvent	EcucChoiceReferenceDef
BSW Description	
Input variable for the programming event, which is assigned to a specific data element used as interface for the Dem-internal PID calculations.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemGeneralOBD
BSW Parameter	**BSW Type**
DemOBDInputVehicleSpeed	EcucChoiceReferenceDef
BSW Description	
Input variable for the vehicle speed, which is assigned to a specific data element used as interface for the Dem-internal PID calculations.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemGeneralOBD
BSW Parameter	**BSW Type**
DemOBDTimeSinceEngineStart	EcucChoiceReferenceDef
BSW Description	

Input variable for the Time Since Engine Start information, which is assigned to a specific data element.
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral

BSW Parameter	**BSW Type**
DemGroupOfDTC	EcucParamConfContainerDef

BSW Description
This container contains the configuration (parameters) for DTC groups.
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemGroupOfDTC

BSW Parameter	**BSW Type**
DemGroupDTCs	EcucIntegerParamDef

BSW Description
DTC values of the selected group of DTC (Range: 3 byte, 0xFFFFFF is reserved for 'all DTCs', according to ISO14229-1 Annex D.1) The DTC group 'all DTCs' is always available and will not be configured. The following ranges are reserved by ISO 14229-1 : 0x000000 to 0x0000ff and 0xffff00 to 0xffffff.
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral

BSW Parameter	**BSW Type**
DemHeaderFileInclusion	EcucStringParamDef

BSW Description

Name of the header file(s) to be included by the Dem module containing the used C-callback declarations.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral
BSW Parameter	**BSW Type**
DemImmediateNvStorageLimit	EcucIntegerParamDef
BSW Description	
This parameter defines the maximum number of occurrences, a specific event memory entry is allowed, to be stored in NVRAM immediately (refer to DemImmediateNvStorage).	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral
BSW Parameter	**BSW Type**
DemIndicator	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) for Indicators.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemIndicator
BSW Parameter	**BSW Type**
DemIndicatorID	EcucIntegerParamDef
BSW Description	
Unique identifier of an indicator.	
Template Description	

M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral
BSW Parameter	**BSW Type**
DemMILIndicatorRef	EcucReferenceDef
BSW Description	
This parameter defines the indicator representing the MIL. This parameter is mandatory for ECUs supporting OBD (refer to DemOBDSupport).	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral
BSW Parameter	**BSW Type**
DemMaxNumberEventEntryEventBuffer	EcucIntegerParamDef
BSW Description	
Specifies the size of the buffer for storing environmental data (freezeframes and extended data) until they are processed and stored to the event memory.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral
BSW Parameter	**BSW Type**
DemMaxNumberEventEntryPermanent	EcucIntegerParamDef
BSW Description	
Maximum number of events which can be stored in the permanent memory. The assignment of an event to this memory type is dynamic and used for emission-related events only.	
Template Description	

M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral

BSW Parameter	**BSW Type**
DemMaxNumberPrestoredFF	EcucIntegerParamDef

BSW Description
Defines the maximum number for prestored freeze frames. If set to 0, then freeze frame prestorage is not supported by the ECU.
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral

BSW Parameter	**BSW Type**
DemMirrorMemory	EcucParamConfContainerDef

BSW Description
This container contains the mirror event memory specific parameters of the Dem module.
Template Description
This represents a mirror memory for a diagnostic event.
M2 Parameter
DiagnosticExtract::Dem::DiagnosticTroubleCode::DiagnosticMemoryDestinationMirror

Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemMirrorMemory

BSW Parameter	**BSW Type**
DemMaxNumberEventEntryMirror	EcucIntegerParamDef

BSW Description
Maximum number of events which can be stored in the mirror memory
Template Description
This attribute fixes the maximum number of event entries in the fault memory.
M2 Parameter
DiagnosticExtract::DiagnosticCommonProps::DiagnosticCommonProps.maxNumberOfEventEntries

Mapping Rule	Mapping Type
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral
BSW Parameter	**BSW Type**
DemNvRamBlockId	EcucParamConfContainerDef
BSW Description	
This container contains the configuration (parameters) for a non-volatile memory block, which is used from the Dem. If no permanent storage of event memory entries is required, no block needs to be configured. The number of blocks which are necessary depends on the implementation and configuration (e.g. number of used event memories) of the Dem module.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemNvRamBlockId
BSW Parameter	**BSW Type**
DemNvRamBlockIdRef	EcucSymbolicNameReferenceDef
BSW Description	
This reference contains the link to a non-volatile memory block. For post build time configurations worst case scenario shall be used.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral
BSW Parameter	**BSW Type**
DemOBDSupport	EcucEnumerationParamDef
BSW Description	
This configuration switch defines OBD support and kind of OBD ECU.	
Template Description	

M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context	
Dem	Dem/DemGeneral	
BSW Parameter		**BSW Type**
DemOccurrenceCounterProcessing		EcucEnumerationParamDef
BSW Description		
This configuration switch defines the consideration of the fault confirmation process for the occurrence counter. For OBD and mixed systems (OBD/non OBD, refer to DemOBDSupport) configuration switch shall always set to DEM_PROCESS_OCCCTR_TF.		
Template Description		
This attribute defines the consideration of the fault confirmation process for the occurrence counter.		
M2 Parameter		
DiagnosticExtract::DiagnosticCommonProps::DiagnosticCommonProps.occurrenceCounterProcessing		
Mapping Rule		**Mapping Type**
1:1 mapping		full
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemGeneral/DemOccurrenceCounterProcessing	
BSW Parameter		**BSW Type**
DEM_PROCESS_OCCCTR_CDTC		EcucEnumerationLiteralDef
BSW Description		
the occurrence counter is triggered by the TestFailed bit if the fault confirmation was successful (ConfirmedDTC bit is set)		
Template Description		
The occurrence counter is triggered by the TestFailed bit if the fault confirmation was successful (ConfirmedDTC bit is set).		
M2 Parameter		
DiagnosticExtract::DiagnosticCommonProps::DiagnosticOccurrenceCounterProcessingEnum.confirmedDtcBit		
Mapping Rule		**Mapping Type**
1:1 mapping		full
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemGeneral/DemOccurrenceCounterProcessing	
BSW Parameter		**BSW Type**
DEM_PROCESS_OCCCTR_TF		EcucEnumerationLiteralDef
BSW Description		
the occurrence counter is only triggered by the TestFailed bit (and the fault confirmation is not considered) This parameter is mandatory in case of J1939.		

Template Description	
The occurrence counter is only triggered by the TestFailed bit (and the fault confirmation is not considered).	
M2 Parameter	
DiagnosticExtract::DiagnosticCommonProps::DiagnosticOccurrenceCounterProcessingEnum.testFailedBit	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral
BSW Parameter	**BSW Type**
DemOperationCycle	EcucParamConfContainerDef
BSW Description	
This container holds all parameters that are relevant to configure an operation cycle.	
Template Description	
Definition of an operation cycle that is the base of the event qualifying and for Dem scheduling.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticOperationCycle::DiagnosticOperationCycle	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemOperationCycle
BSW Parameter	**BSW Type**
DemOperationCycleAutomaticEnd	EcucBooleanParamDef
BSW Description	
If DemOperationCycleAutomaticEnd is configured to TRUE, Dem shall automatically end the driving cycle at either Dem_Shutdown() or Dem_Init().	
Template Description	
If set to true the driving cycle shall automatically end at either Dem_Shutdown() or Dem_Init().	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticOperationCycle::DiagnosticOperationCycle.automaticEnd	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemOperationCycle
BSW Parameter	**BSW Type**
DemOperationCycleAutostart	EcucBooleanParamDef
BSW Description	
The autostart property defines if the operation cycles is automatically (re-)started during Dem_PreInit.	
Template Description	

This attribute defines if the operation cycles is automatically re-started during Dem_PreInit.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticOperationCycle::DiagnosticOperationCycle.cycleAutostart	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemOperationCycle
BSW Parameter	**BSW Type**
DemOperationCycleId	EcucIntegerParamDef
BSW Description	
This parameter's value is used, together with the aggregating container, to define a symbolic name of the operation cycle.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemOperationCycle
BSW Parameter	**BSW Type**
DemOperationCycleType	EcucEnumerationParamDef
BSW Description	
Operation cycles types for the Dem to be supported by cycle-state APIs. Further cycle types can be specified as part of the Dem delivery.	
Template Description	
Operation cycles types for the Dem to be supported by cycle-state APIs.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticOperationCycle::DiagnosticOperationCycle.type	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemOperationCycle/DemOperationCycleType
BSW Parameter	**BSW Type**
DEM_OPCYC_IGNITION	EcucEnumerationLiteralDef
BSW Description	
Ignition ON / OFF cycle	
Template Description	
Ignition ON / OFF cycle	
M2 Parameter	

DiagnosticExtract::Dem::DiagnosticOperationCycle::DiagnosticOperationCycleTypeEnum.ignition	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemOperationCycle/DemOperationCycleType
BSW Parameter	**BSW Type**
DEM_OPCYC_OBD_DCY	EcucEnumerationLiteralDef
BSW Description	
OBD Driving cycle	
Template Description	
OBD Driving cycle	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticOperationCycle::DiagnosticOperationCycleTypeEnum.obd DrivingCycle	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemOperationCycle/DemOperationCycleType
BSW Parameter	**BSW Type**
DEM_OPCYC_OTHER	EcucEnumerationLiteralDef
BSW Description	
further operation cycle	
Template Description	
further operation cycle	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticOperationCycle::DiagnosticOperationCycleTypeEnum.other	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemOperationCycle/DemOperationCycleType
BSW Parameter	**BSW Type**
DEM_OPCYC_TIME	EcucEnumerationLiteralDef
BSW Description	
Time based operation cycle	
Template Description	
Time based operation cycle	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticOperationCycle::DiagnosticOperationCycleTypeEnum.time	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**

valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemOperationCycle/DemOperationCycleType
BSW Parameter	**BSW Type**
DEM_OPCYC_WARMUP	EcucEnumerationLiteralDef
BSW Description	
OBD OBD Warm up cycle	
Template Description	
OBD Warm up cycle	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticOperationCycle::DiagnosticOperationCycleTypeEnum.warmup	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral
BSW Parameter	**BSW Type**
DemOperationCycleStatusStorage	EcucBooleanParamDef
BSW Description	
Defines if the operation cycle state is available over the power cycle (stored non-volatile) or not. true: the operation cycle state is stored non-volatile false: the operation cycle state is only stored volatile	
Template Description	
Defines if the operation cycle state is available over the power cycle (stored non-volatile) or not. true: the operation cycle state is stored non-volatile false: the operation cycle state is only stored volatile	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticOperationCycle::DiagnosticOperationCycle.cycleStatusStorage	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral
BSW Parameter	**BSW Type**
DemPTOSupport	EcucBooleanParamDef
BSW Description	
This configuration switch defines, whether PTO support (and therefore PID $1E support) is enabled or not.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**

Mapping Status	Mapping ID
valid	

BSW Module	BSW Context	
Dem	Dem/DemGeneral	
BSW Parameter	**BSW Type**	
DemPrimaryMemory	EcucParamConfContainerDef	
BSW Description		
This container contains the primary event memory specific parameters of the Dem module.		
Template Description		
This represents a primary memory for a diagnostic event.		
M2 Parameter		
DiagnosticExtract::Dem::DiagnosticTroubleCode::DiagnosticMemoryDestinationPrimary		
Mapping Rule		**Mapping Type**
1:1 mapping		full
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemGeneral/DemPrimaryMemory	
BSW Parameter	**BSW Type**	
DemMaxNumberEventEntryPrimary	EcucIntegerParamDef	
BSW Description		
Maximum number of events which can be stored in the primary memory		
Template Description		
This attribute fixes the maximum number of event entries in the fault memory.		
M2 Parameter		
DiagnosticExtract::DiagnosticCommonProps::DiagnosticCommonProps.maxNumberOfEventEntries		
Mapping Rule		**Mapping Type**
1:1 mapping		full
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemGeneral	
BSW Parameter	**BSW Type**	
DemRatio	EcucParamConfContainerDef	
BSW Description		
This container contains the OBD-specific in-use-monitor performance ratio configuration. It is related to a specific event, a FID, and an IUMPR group.		
Template Description		
Specifies the abstract needs of a component or module on the configuration of OBD Services in relation to a particular "ratio monitoring" which is supported by this component or module.		
M2 Parameter		
CommonStructure::ServiceNeeds::ObdRatioServiceNeeds		
Mapping Rule		**Mapping Type**
In case the owner of the ObdRatioServiceNeeds is a BSW module then the DemRatio.shortName = {capitalizedMip}_{ServiceDependency.symbolicNameProps.symbol}.		full

Mapping Status	Mapping ID
valid	up_Dem_00001

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemRatio
BSW Parameter	**BSW Type**
DemDiagnosticEventRef	EcucReferenceDef
BSW Description	
This reference contains the link to a diagnostic event.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemRatio
BSW Parameter	**BSW Type**
DemFunctionIdRef	EcucSymbolicNameReferenceDef
BSW Description	
This reference contains the link to a function identifier within the FiM which is used as a primary FID.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemRatio
BSW Parameter	**BSW Type**
DemIUMPRDenGroup	EcucEnumerationParamDef
BSW Description	
This parameter specifies the assigned denominator type which is applied in addition to the General Denominator conditions.	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context	
Dem	Dem/DemGeneral/DemRatio	
BSW Parameter		**BSW Type**
DemIUMPRGroup		EcucEnumerationParamDef
BSW Description		
This parameter specifies the assigned IUMPR group of the ratio Id.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemGeneral/DemRatio	
BSW Parameter		**BSW Type**
DemRatioId		EcucIntegerParamDef
BSW Description		
Defines a unique ratio Id. This parameter should not be changeable by user, because the Id should be generated by Dem itself to prevent gaps and multiple use of an Id. The ratio Ids should be sequentially ordered beginning with 0 and no gaps in between.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemGeneral/DemRatio	
BSW Parameter		**BSW Type**
DemRatioKind		EcucEnumerationParamDef
BSW Description		
This parameter defines whether the ratio will be calculated API or observer based.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context

Dem	Dem/DemGeneral/DemRatio	
BSW Parameter		**BSW Type**
DemSecondaryFunctionIdRef		EcucSymbolicNameReferenceDef
BSW Description		
This reference contains the link to a function identifier within the FiM which is used as a secondary FID. The "primary" and all "secondary" FID inhibitions are combined by "OR".		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	**BSW Context**	
Dem	Dem/DemGeneral	
BSW Parameter		**BSW Type**
DemResetConfirmedBitOnOverflow		EcucBooleanParamDef
BSW Description		
This configuration switch defines, whether the confirmed bit is reset or not while an event memory entry will be displaced.		
Template Description		
This attribute defines, whether the confirmed bit is reset or not while an event memory entry will be displaced.		
M2 Parameter		
DiagnosticExtract::DiagnosticCommonProps::DiagnosticCommonProps.resetConfirmedBitOn Overflow		
Mapping Rule		**Mapping Type**
1:1 mapping		full
Mapping Status		**Mapping ID**
valid		

BSW Module	**BSW Context**	
Dem	Dem/DemGeneral	
BSW Parameter		**BSW Type**
DemStatusBitHandlingTestFailedSinceLastClear		EcucEnumerationParamDef
BSW Description		
This configuration switch defines, whether the aging and displacement mechanism shall be applied to the "TestFailedSinceLastClear" status bits.		
Template Description		
This attribute defines, whether the aging and displacement mechanism shall be applied to the "Test-FailedSinceLastClear" status bits.		
M2 Parameter		
DiagnosticExtract::DiagnosticCommonProps::DiagnosticCommonProps.statusBitHandlingTest FailedSinceLastClear		
Mapping Rule		**Mapping Type**
1:1 mapping		full
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemGeneral	
BSW Parameter		**BSW Type**
DemStatusBitStorageTestFailed		EcucBooleanParamDef
BSW Description		
Activate/Deactivate the permanent storage of the "TestFailed" status bits. true: storage activated false: storage deactivated		
Template Description		
This parameter is used to activate/deactivate the permanent storage of the "TestFailed" status bits. true: storage activated false: storage deactivated		
M2 Parameter		
DiagnosticExtract::DiagnosticCommonProps::DiagnosticCommonProps.statusBitStorageTestFailed		
Mapping Rule		**Mapping Type**
1:1 mapping		full
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemGeneral	
BSW Parameter		**BSW Type**
DemStorageCondition		EcucParamConfContainerDef
BSW Description		
This container contains the configuration (parameters) for storage conditions.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemGeneral/DemStorageCondition	
BSW Parameter		**BSW Type**
DemStorageConditionId		EcucIntegerParamDef
BSW Description		
Defines a unique storage condition Id. This parameter should not be changeable by user, because the Id should be generated by Dem itself to prevent gaps and multiple use of an Id. The storage conditions should be sequentially ordered beginning with 0 and no gaps in between.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemGeneral/DemStorageCondition	
BSW Parameter		**BSW Type**
DemStorageConditionReplacementEventRef		EcucReferenceDef
BSW Description		
Specifies the reference to an event which is stored to event memory and supports failure analysis.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemGeneral/DemStorageCondition	
BSW Parameter		**BSW Type**
DemStorageConditionStatus		EcucBooleanParamDef
BSW Description		
Defines the initial status for enable or disable of storage of a diagnostic event. The value is the initialization after power up (before this condition is reported the first time). true: storage of a diagnostic event enabled false: storage of a diagnostic event disabled		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context	
Dem	Dem/DemGeneral	
BSW Parameter		**BSW Type**
DemStorageConditionGroup		EcucParamConfContainerDef
BSW Description		
This container contains the configuration (parameters) for storage condition groups.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	BSW Context

Dem	Dem/DemGeneral/DemStorageConditionGroup	
BSW Parameter		**BSW Type**
DemStorageConditionRef		EcucReferenceDef
BSW Description		
References an enable condition.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	**BSW Context**	
Dem	Dem/DemGeneral	
BSW Parameter		**BSW Type**
DemSuppressionSupport		EcucEnumerationParamDef
BSW Description		
This configuration switch defines, whether support for suppression is enabled or not.		
Template Description		
M2 Parameter		
Mapping Rule		**Mapping Type**
Mapping Status		**Mapping ID**
valid		

BSW Module	**BSW Context**	
Dem	Dem/DemGeneral	
BSW Parameter		**BSW Type**
DemTaskTime		EcucFloatParamDef
BSW Description		

Allow to configure the time for the periodic cyclic task. Please note: This configuration value shall be equal to the value in the Basic Software Scheduler configuration of the RTE module. The AUTOSAR configuration standard is to use SI units, so this parameter is defined as float value in seconds. Dem configuration tools must convert this float value to the appropriate value format for the use in the software implementation of Dem. min: A negative value is not allowed. max: After event status was reported, processing shall be completed within 100ms in order to have the fault entry status information updated as soon as possible (e.g. for PID $01). upperMultiplicity: Exactly one TaskTime must be specified per configuration. lowerMultiplicity: Exactly one TaskTime must be specified per configuration.
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral

BSW Parameter	**BSW Type**
DemTriggerDcmReports	EcucBooleanParamDef

BSW Description
Activate/Deactivate the notification to the Diagnostic Communication Manager for ROE processing. true: Dcm ROE notification activated false: Dcm ROE notification deactivated
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral

BSW Parameter	**BSW Type**
DemTriggerDltReports	EcucBooleanParamDef

BSW Description

Activate/Deactivate the notification to the Diagnostic Log and Trace. true: Dlt notification activated false: Dlt notification deactivated
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral

BSW Parameter	**BSW Type**
DemTriggerFiMReports	EcucBooleanParamDef

BSW Description
Activate/Deactivate the notification to the Function Inhibition Manager. true: FiM notification activated false: FiM notification deactivated
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral

BSW Parameter	**BSW Type**
DemTriggerMonitorInitBeforeClearOk	EcucBooleanParamDef

BSW Description
Defines if the monitor re-initialization has to be triggered before or after the Dem module returns DEM_CLEAR_OK. true: trigger re-initialization before DEM_CLEAR_OK false: trigger re-initialization after DEM_CLEAR_OK
Template Description
M2 Parameter

Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral

BSW Parameter	BSW Type
DemTypeOfDTCSupported	EcucEnumerationParamDef
BSW Description	
This parameter defines the format returned by Dem_DcmGetTranslationType and does not relate to/influence the supported Dem functionality.	
Template Description	
This attribute defines the format returned by Dem_DcmGetTranslationType and does not relate to/influence the supported Dem functionality.	
M2 Parameter	
DiagnosticExtract::DiagnosticCommonProps::DiagnosticCommonProps.typeOfDtcSupported	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemTypeOfDTCSupported
BSW Parameter	**BSW Type**
DEM_DTC_TRANSLATION_ISO11992_4	EcucEnumerationLiteralDef
BSW Description	
ISO11992-4 DTC format	
Template Description	
ISO11992-4 DTC format	
M2 Parameter	
DiagnosticExtract::DiagnosticCommonProps::DiagnosticTypeOfDtcSupportedEnum.iso11992_4	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemTypeOfDTCSupported
BSW Parameter	**BSW Type**
DEM_DTC_TRANSLATION_ISO14229_1	EcucEnumerationLiteralDef
BSW Description	
ISO14229-1 DTC format (3 byte format)	
Template Description	
ISO14229-1 DTC format (3 byte format)	
M2 Parameter	
DiagnosticExtract::DiagnosticCommonProps::DiagnosticTypeOfDtcSupportedEnum.iso14229_1	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemTypeOfDTCSupported
BSW Parameter	**BSW Type**
DEM_DTC_TRANSLATION_SAEJ1939_73	EcucEnumerationLiteralDef
BSW Description	

SAEJ1939-73 DTC format	
Template Description	
SAEJ1939-73 DTC format	
M2 Parameter	
DiagnosticExtract::DiagnosticCommonProps::DiagnosticTypeOfDtcSupportedEnum.saeJ1939_73	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemTypeOfDTCSupported
BSW Parameter	**BSW Type**
DEM_DTC_TRANSLATION_SAE_J2012_DA_DTCFORMAT_04	EcucEnumerationLiteralDef
BSW Description	
SAE_J2012-DA_DTCFormat_00 (3 byte format)	
Template Description	
SAE_J2012-DA_DTCFormat_00 (3 byte format)	
M2 Parameter	
DiagnosticExtract::DiagnosticCommonProps::DiagnosticTypeOfDtcSupportedEnum.saeJ2012_da	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral
BSW Parameter	**BSW Type**
DemTypeOfFreezeFrameRecordNumeration	EcucEnumerationParamDef
BSW Description	
This parameter defines the type of assigning freeze frame record numbers for event-specific freeze frame records.	
Template Description	
This attribute defines the type of assigning freeze frame record numbers for event-specific freeze frame records.	
M2 Parameter	
DiagnosticExtract::DiagnosticCommonProps::DiagnosticCommonProps.typeOfFreezeFrameRecordNumeration	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemTypeOfFreezeFrameRecordNumeration
BSW Parameter	**BSW Type**
DEM_FF_RECNUM_CALCULATED	EcucEnumerationLiteralDef
BSW Description	
freeze frame records will be numbered consecutive starting by 1 in their chronological order	

Template Description	
Freeze frame records will be numbered consecutive starting by 1 in their chronological order.	
M2 Parameter	
DiagnosticExtract::DiagnosticCommonProps::DiagnosticTypeOfFreezeFrameRecordNumerationEnum.calculated	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemTypeOfFreezeFrameRecordNumeration
BSW Parameter	**BSW Type**
DEM_FF_RECNUM_CONFIGURED	EcucEnumerationLiteralDef
BSW Description	
freeze frame records will be numbered based on the given configuration in their chronological order	
Template Description	
Freeze frame records will be numbered based on the given configuration in their chronological order.	
M2 Parameter	
DiagnosticExtract::DiagnosticCommonProps::DiagnosticTypeOfFreezeFrameRecordNumerationEnum.configured	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral
BSW Parameter	**BSW Type**
DemUserDefinedMemory	EcucParamConfContainerDef
BSW Description	
This container contains the user defined event memory specific parameters of the Dem module.	
Template Description	
This represents a user-defined memory for a diagnostic event.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticTroubleCode::DiagnosticMemoryDestinationUserDefined	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	BSW Context
Dem	Dem/DemGeneral/DemUserDefinedMemory
BSW Parameter	**BSW Type**
DemMaxNumberEventEntryUserDefined	EcucIntegerParamDef
BSW Description	
Maximum number of events which can be stored in the user defined memory.	
Template Description	
This attribute fixes the maximum number of event entries in the fault memory.	
M2 Parameter	

DiagnosticExtract::DiagnosticCommonProps::DiagnosticCommonProps.maxNumberOfEventEntries	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral/DemUserDefinedMemory
BSW Parameter	**BSW Type**
DemUserDefinedMemoryIdentifier	EcucIntegerParamDef
BSW Description	
Identifier used by external tester to identify the User defined event memory.	
Template Description	
This represents the identifier of the user-defined memory.	
M2 Parameter	
DiagnosticExtract::Dem::DiagnosticTroubleCode::DiagnosticMemoryDestinationUserDefined.memoryId	
Mapping Rule	**Mapping Type**
1:1 mapping	full
Mapping Status	**Mapping ID**
valid	

BSW Module	**BSW Context**
Dem	Dem/DemGeneral
BSW Parameter	**BSW Type**
DemVersionInfoApi	EcucBooleanParamDef
BSW Description	
Activate/Deactivate the version information API. true: version information activated false: version information deactivated	
Template Description	
M2 Parameter	
Mapping Rule	**Mapping Type**
Mapping Status	**Mapping ID**
valid	

F Splitable Elements in the Scope of this Document

This chapter contains a table of all model elements stereotyped ≪atpSplitable≫ in the scope of this document.

Each entry in Table F.1 consists of the identification of the specific model element itself and the applicable value of the tagged value atp.Splitkey.

For more information about the concept of splitable model elements and how these shall be treated please refer to [19].

Name of splitable element	Splitkey
DiagnosticAging.agingCycle	agingCycle, variationPoint.ShortLabel
DiagnosticContributionSet.commonProperties	commonProperties
DiagnosticContributionSet.element	element, variationPoint.shortLabel
DiagnosticContributionSet.serviceTable	serviceTable, variationPoint.shortLabel
DiagnosticDataIdentifier.dataElement	shortName, variationPoint.shortLabel
DiagnosticEnableConditionGroup.enableCondition	enableCondition, variationPoint.shortLabel
DiagnosticEvent.connectedIndicator	shortName, variationPoint.shortLabel
DiagnosticParameter.dataElement	shortName, variationPoint.shortLabel
DiagnosticSecurityAccess.securityLevel	securityLevel
DiagnosticServiceTable.diagnosticConnection	diagnosticConnection, variationPoint.shortLabel
DiagnosticStorageConditionGroup.storageCondition	storageCondition, variationPoint.shortLabel
DiagnosticTroubleCodeGroup.dtc	dtc, variationPoint.shortLabel
DiagnosticTroubleCodeProps.extendedDataRecord	shortName, variationPoint.shortLabel
DiagnosticTroubleCodeProps.freezeFrame	shortName, variationPoint.shortLabel

Table F.1: Usage of splitable elements

G Variation Points in the Scope of this Document

This chapter contains a table of all model elements stereotyped ≪atpVariation≫ in the scope of this document.

Each entry in Table G.1 consists of the identification of the model element itself and the applicable value of the tagged value vh.latestBindingTime.

For more information about the concept of variation points and how model elements that contain variation points shall be treated please refer to [19].

Variation Point	Latest Binding Time
DiagnosticAging.agingCycle	preCompileTime
DiagnosticAging.threshold	preCompileTime
DiagnosticContributionSet.element	postBuild
DiagnosticContributionSet.serviceTable	postBuild
DiagnosticDataIdentifier.dataElement	postBuild
DiagnosticEnableConditionGroup.enableCondition	postBuild
DiagnosticEvent.connectedIndicator	postBuild
DiagnosticEvent.eventFailureCycleCounterThreshold	postBuild
DiagnosticFreezeFrame.recordNumber	preCompileTime
DiagnosticIndicator.healingCycleCounterThreshold	preCompileTime
DiagnosticParameter.dataElement	postBuild
DiagnosticRoutine.id	preCompileTime
DiagnosticServiceTable.diagnosticConnection	postBuild
DiagnosticStorageConditionGroup.storageCondition	postBuild
DiagnosticTroubleCodeGroup.dtc	postBuild
DiagnosticTroubleCodeGroup.groupNumber	preCompileTime
DiagnosticTroubleCodeJ1939.j1939DtcValue	preCompileTime
DiagnosticTroubleCodeObd.considerPtoStatus	preCompileTime
DiagnosticTroubleCodeObd.obdDTCValue	preCompileTime
DiagnosticTroubleCodeProps.extendedDataRecord	preCompileTime
DiagnosticTroubleCodeProps.freezeFrame	preCompileTime

Table G.1: Usage of variation points