Document Title	Specification of Data Distribution Service in Classic Platform
Document Owner	AUTOSAR
Document Responsibility	AUTOSAR
Document Identification No	1069

Document Status	published
Part of AUTOSAR Standard	Classic Platform
Part of Standard Release	R22-11

Document Change History			
Date	**Release**	**Changed by**	**Description**
2022-11-24	R22-11	AUTOSAR Release Management	• Initial release

Disclaimer

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

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

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

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

The word AUTOSAR and the AUTOSAR logo are registered trademarks.

Contents

Known Limitations

None.

1 Introduction and functional overview

This specification specifies the functionality, API and the configuration of the **Data Distribution Service AUTOSAR Basic Software module (Dds BSW)** .

1.1 DDS protocol overview

The **Data Distribution Service (DDS)** is a middleware protocol and API standard from the Object Management Group (OMG).

A preliminary overview of DDS can be found into chapter 4 of AUTOSAR_RS_DDS or referring directly the OMG standard [1].

2 Acronyms and Abbreviations

2.1 Acronyms

For acronyms and abbreviations refer to [2, AUTOSAR glossary].

2.2 Abbreviations

None.

3 Related documentation

3.1 Input documents & related standards and norms

[1] Data Distribution Service (DDS), Version 1.4
http://www.omg.org/spec/DDS/1.4

[2] Glossary
AUTOSAR_TR_Glossary

[3] General Specification of Basic Software Modules
AUTOSAR_SWS_BSWGeneral

[4] Specification of RTE Software
AUTOSAR_SWS_RTE

[5] Specification of PDU Router
AUTOSAR_SWS_PDURouter

[6] DDS Interoperability Wire Protocol, Version 2.2
http://www.omg.org/spec/DDSI-RTPS/2.2

[7] Specification of Default Error Tracer
AUTOSAR_SWS_DefaultErrorTracer

[8] Specification of Crypto Service Manager
AUTOSAR_SWS_CryptoServiceManager

[9] Specification of CRC Routines
AUTOSAR_SWS_CRCLibrary

[10] Requirements on Data Distribution Service
AUTOSAR_RS_DataDistributionService

[11] DDS Security, Version 1.1
https://www.omg.org/spec/DDS-SECURITY/1.1

[12] ISO 26262:2018 (all parts) – Road vehicles – Functional Safety
http://www.iso.org

[13] Extensible and Dynamic Topic Types for DDS, Version 1.2
https://www.omg.org/spec/DDS-XTypes/1.2

[14] Specification of Platform Types
AUTOSAR_SWS_PlatformTypes

3.2 Related specification

AUTOSAR provides a General Specification on Basic Software modules [3], which is also valid for the Dds BSW.

Thus, the specification SWS BSW General shall be considered as additional and required specification for Dds.

4 Constraints and assumptions

4.1 Constraints and assumptions

4.1.1 Assumptions

- **DDS/non DDS PDU concatenation.** Concatenating DDS PDU and non-DDS PDU by IpduM is out of the scope of this concept.

4.1.2 Limitations

- **DDS Dynamic Discovery [1]:** Not supported.
- **Service Oriented Architecture (SOA):** Not supported.
- **AP/CP Interoperability by means of SOA communication:** Not supported.
- **DDS/CAN Data gateway:** Not supported
- **System configuration:** Not supported.
- **Shared-memory communication:** Not supported.
- **Immediate transmission:** Not supported.
- **Immediate reception:** Not supported.
- **Dds_TriggerTransmit:** Only supported if called by lower-layer.
- **Communication use cases:** Below a table summarizing the supported Communication paths.

Short name:	Y/N
Signal-based	Yes
Service-oriented	No
Diagnostic	No
DLT	No
XCP	No
Mirror	No

Table 4.1: DDS supported Communication paths

4.1.3 Constraints

- **DDS Transformer:** Since DDS needs to access to data and data type, no transformation has to be performed at RTE level.
- **DDS-SOME/IP network interaction:** Dds and SOME/IP share the same bus type, so some mechanism to assure they don't interfere would be provided. This

mechanism will mostly consist on UDP port choice: SOME/IP and DDS shall never share reception port numbers.

- **DDS-SOME/IP SoAd resource sharing:** Shall not be mix of DDS and SOME/IP (or any other potential protocol) communication on the same socket connection. A socket (or a set of sockets) is reserved for DDS only.
- **DDS/non DDS PDU concatenation:** DDS and SOME/IP shall not share the same socket connection. A socket (or a set of sockets) shall be reserved to DDS only.
- **UDP Usage:** According to the OMG specification [1], the UDP/IP PSM shall be used for inter-ecu communications.

4.2 Applicability to car domains

This module is applicable all domains where DDS communication is required and/or benefitial.

5 Dependencies to other modules

This section describes the relations to other modules and files within the AUTOSAR basic software architecture. It contains brief descriptions of configuration information and services, which are required by the Dds module from other modules.

5.1 RTE (BSW Scheduler)

The RTE BSW Scheduler [4] calls the main functions of the Dds BSW, which are necessary for the cyclic processes of the Dds.

5.2 Pdu Router

The Dds module uses the PduR [5] as middle layer module.

5.3 StbM

OMG Standard states that each RTPS message sent by the originating Participant can include a timestamp. ([6] 8.3.2.2). It may be used by the receiving application to estimate the time offset between the clocks of the sending and receiving Participants (for instance in DESTINATION_ORDER QoS policy handling). The StbM_GetCurrentTime() API shall be used to guarantee the needed precision ([6] 8.3.5.8, 9.4.2.9).

5.4 Default Error Tracer

In order to be able to report development errors, the Dds module has to have access to the error hook of the Default Error Tracer [7].

5.5 Crypto Service Manager

In order to support Security capabilities (e.g. Key management, Message Authentication Code generation and verification), the Dds shall use the Crypto Service Module API [8]: The Dds requires:

- the MAC-generate interface *(Csm_MacGenerate)* to generate MAC to be added to messages to be sent;

- the MAC-verify interface *(Csm_MacVerify)* to check MAC of received messages.

5.6 Cyclic Redundancy Check

In order to support Safety capabilities, Dds uses the CRC Library [9].

The Dds requires the *Crc_CalculateCRC32* or the *Crc_CalculateCRC64* APIs to calculate CRC to be added to messages to be sent or to be checked for received messages.

6 Requirements Tracing

The following tables reference the requirements specified in [10, RS-DDS] and links to the fulfillment of these. Please note that if column “Satisfied by” is empty for a specific requirement this means that this requirement is not fulfilled by this document.

Requirement	Description	Satisfied by
[FO_RS_Dds_00001]	DDS Compliance	[CP_SWS_Dds_00734] [CP_SWS_Dds_CONSTR_00725]
[FO_RS_Dds_00002]	DDS standard serialization rules	[CP_SWS_Dds_00726]
[FO_RS_Dds_00004]	DDS payload serialization rules	[CP_SWS_Dds_00727] [CP_SWS_Dds_00728] [CP_SWS_Dds_00729] [CP_SWS_Dds_00730] [CP_SWS_Dds_00731] [CP_SWS_Dds_00734] [CP_SWS_Dds_00735] [CP_SWS_Dds_CONSTR_00725] [CP_SWS_Dds_CONSTR_00732] [CP_SWS_Dds_CONSTR_00733]
[FO_RS_Dds_00005]	DDS Quality of Service	[CP_SWS_Dds_00763] [CP_SWS_Dds_01001]
[FO_RS_Dds_00007]	Type Definition	[CP_SWS_Dds_00727] [CP_SWS_Dds_00728] [CP_SWS_Dds_00729] [CP_SWS_Dds_00730] [CP_SWS_Dds_00731] [CP_SWS_Dds_00735] [CP_SWS_Dds_CONSTR_00732] [CP_SWS_Dds_CONSTR_00733]
[FO_RS_Dds_00008]	Customization	[CP_SWS_Dds_CONSTR_00712]
[FO_RS_Dds_00009]	Security mechanism	[CP_SWS_Dds_00750] [CP_SWS_Dds_00752] [CP_SWS_Dds_00753] [CP_SWS_Dds_00756] [CP_SWS_Dds_00758] [CP_SWS_Dds_CONSTR_00743] [CP_SWS_Dds_CONSTR_00754]
[FO_RS_Dds_00010]	Safety mechanism	[CP_SWS_Dds_00761] [CP_SWS_Dds_00762] [CP_SWS_Dds_00763] [CP_SWS_Dds_00766] [CP_SWS_Dds_00769]
[FO_RS_Dds_00015]	Publish	[CP_SWS_Dds_00829] [CP_SWS_Dds_00830] [CP_SWS_Dds_00851] [CP_SWS_Dds_00852] [CP_SWS_Dds_00854] [CP_SWS_Dds_00855] [CP_SWS_Dds_00871] [CP_SWS_Dds_00872] [CP_SWS_Dds_00881] [CP_SWS_Dds_00882] [CP_SWS_Dds_00883] [CP_SWS_Dds_00884]
[FO_RS_Dds_00016]	Subscribe	[CP_SWS_Dds_00825] [CP_SWS_Dds_00826] [CP_SWS_Dds_00827] [CP_SWS_Dds_00828] [CP_SWS_Dds_00861] [CP_SWS_Dds_00862] [CP_SWS_Dds_00863] [CP_SWS_Dds_00864]
[FO_RS_Dds_00017]	Transport protocol	[CP_SWS_Dds_00726]
[FO_RS_Dds_00019]	RTPS message encapsulation	[CP_SWS_Dds_00726] [CP_SWS_Dds_00727] [CP_SWS_Dds_00734]
[FO_RS_Dds_00020]	RTPS message decapsulation	[CP_SWS_Dds_00726] [CP_SWS_Dds_00727] [CP_SWS_Dds_00734]
[FO_RS_Dds_-CONSTR_00006]	The DDS AUTOSAR components receive unserialized data	[CP_SWS_Dds_CONSTR_00725]
[SRS_BSW_00003]	All software modules shall provide version and identification information	[CP_SWS_Dds_00820] [CP_SWS_Dds_00821] [CP_SWS_Dds_00831]
[SRS_BSW_00101]	The Basic Software Module shall be able to initialize variables and hardware in a separate initialization function	[CP_SWS_Dds_00810] [CP_SWS_Dds_00811] [CP_SWS_Dds_00812] [CP_SWS_Dds_00813]

▽

△

Requirement	Description	Satisfied by
[SRS_BSW_00318]	Each AUTOSAR Basic Software Module file shall provide version numbers in the header file	[CP_SWS_Dds_00820] [CP_SWS_Dds_00831]
[SRS_BSW_00373]	The main processing function of each AUTOSAR Basic Software Module shall be named according the defined convention	[CP_SWS_Dds_00823] [CP_SWS_Dds_00824]
[SRS_BSW_00374]	All Basic Software Modules shall provide a readable module vendor identification	[CP_SWS_Dds_00820] [CP_SWS_Dds_00831]
[SRS_BSW_00379]	All software modules shall provide a module identifier in the header file and in the module XML description file.	[CP_SWS_Dds_00820] [CP_SWS_Dds_00831]
[SRS_BSW_00402]	Each module shall provide version information	[CP_SWS_Dds_00820] [CP_SWS_Dds_00831]
[SRS_BSW_00405]	BSW Modules shall support multiple configuration sets	[CP_SWS_Dds_00810]
[SRS_BSW_00407]	Each BSW module shall provide a function to read out the version information of a dedicated module implementation	[CP_SWS_Dds_00820] [CP_SWS_Dds_00831]
[SRS_BSW_00411]	All AUTOSAR Basic Software Modules shall apply a naming rule for enabling/disabling the existence of the API	[CP_SWS_Dds_00820] [CP_SWS_Dds_00831]
[SRS_BSW_00414]	Init functions shall have a pointer to a configuration structure as single parameter	[CP_SWS_Dds_00810]
[SRS_BSW_00424]	BSW module main processing functions shall not be allowed to enter a wait state	[CP_SWS_Dds_00823] [CP_SWS_Dds_00824]
[SRS_BSW_00433]	Main processing functions are only allowed to be called from task bodies provided by the BSW Scheduler	[CP_SWS_Dds_00823] [CP_SWS_Dds_00824]

Table 6.1: RequirementsTracing

7 Functional specification

7.1 Overview

The Dds module implements all the interface logic (Entity management, QoS, etc.) and the DDSI-RTPS standard layer [6] for DDS communication. It is a full-fledged middleware composed by several functional aspects:

- Serialization
- Data filtering
- Data reordering
- Data persistency
- Data re-transmission
- Security
- E2E protection
- PDU Router interaction

From the transmission path point of view, Dds interacts with the Pdu Router only offering a PDU-based interface for the incoming (e.g. **upper layer PDU**) and outgoing (e.g. **lower layer PDU**) PDUs. Basically, upper layer PDU contains the unserialised data coming from the application layer, by means of the Rte, LdCom, PduR. The Dds module is able to process the data through its **type** mapped to the PDU (see 10.2). The lower layer PDU contains the DDSI-RTPS protocol packet ready to be delivered to the transport layer.

The **transport layer** provides a set connections suitable to enable the Dds communication. For instance, let's consider a simple publishing SW-C using some Publishers/DataWriters under some DomainParticipants. There is no need to create the Subscribers/DataReaders configuration on local Ecu at Dds level, but transport layer shall guarantee they are reachable by the lower PDUs. Similar thing shall happen at the receiving side, the remote transport configuration shall guarantee that received PDUs reach the proper DataReaders/Subscribers and subscribing applications.

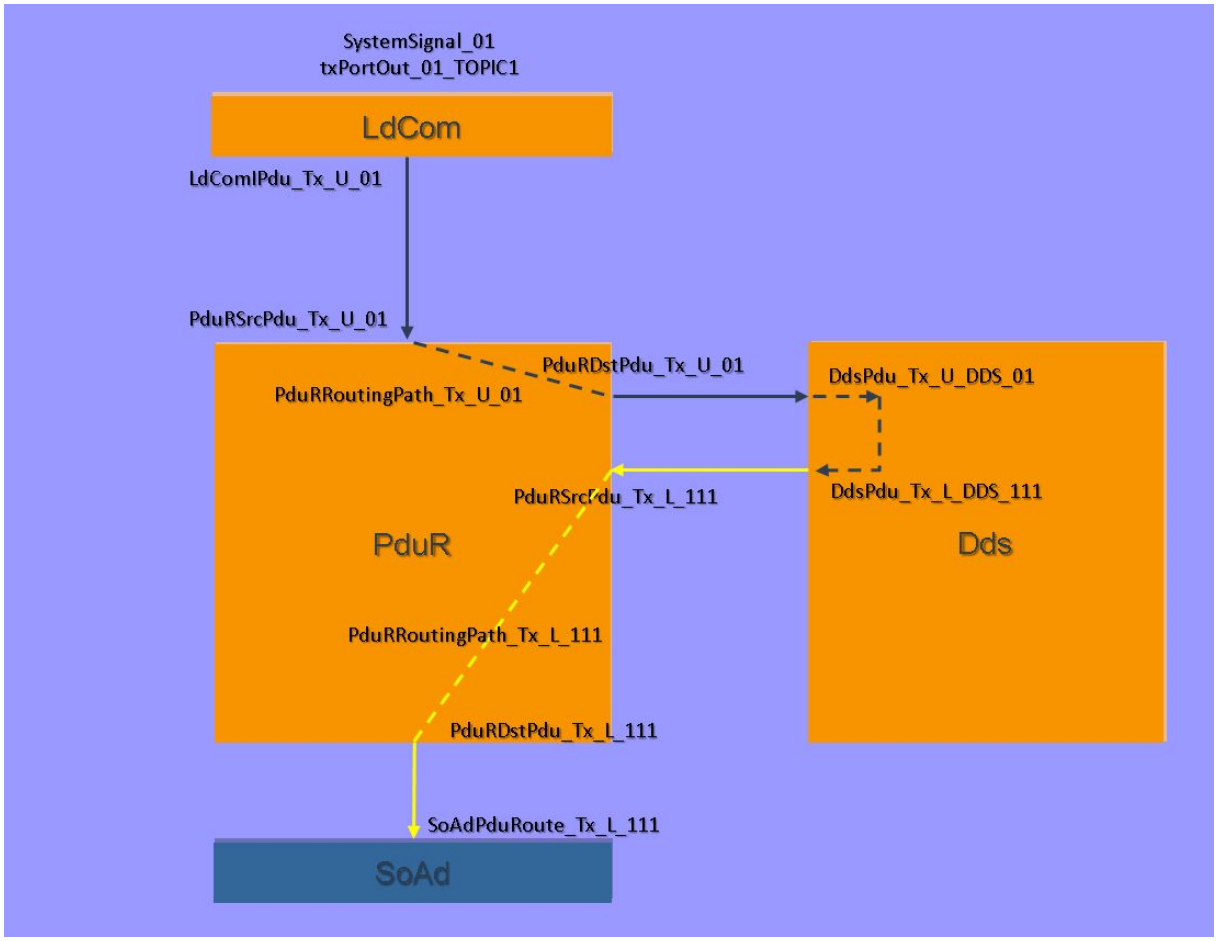

Figure 7.1: Dds full transmission path

7.1.1 QoS Management

The Dds BSW could support a subset (even empty) of QoS policies. There is no mandatory QoS to be implemented to be still compliant: which QoS policy is actually supported is vendor-specific.

Each entity could define its own set of supported QoS policy, by static configuration.

7.1.2 Security Mechanisms

Opening up a communication path between AP and CP, and even between CP and non-AUTOSAR platforms, could involve security risks, so it may require the usage of some security mechanisms.

The Dds BSW Module guarantees some security mechanism by the usage of DDS Security Specification [11]. The usage of this specification is necessary to guarantee the interoperability with other DDS systems, both with AP (where DDS- Security is already in use) and in non-AUTOSAR systems. Implementing this specification, however,

could be really resource consuming. In particular, to be used on a slow microcontroller, these features would need hardware acceleration. To overcome to this issue, a subset of DDS-security functionalities which guarantee a minimum security level has been selected.

At this stage, implementing DDS-Security aims to guarantee message authentication, data integrity and group authentication. Security mechanism can be enabled or disabled at configuration time. If enabled, all security parameters must been statically configured at pre-compile time. For details on security parameters configuration, please refer to paragraph 10.2.3.2.2.

If configured, a Message Authentication Code (MAC) of the entire RTPS message is added. The AUTOSAR CSM is used for key management and MAC calculation. Which algorithm to be used is configurable (choosing from supported ones).

The key used for hash algorithms are symmetric keys shared between entities associated to a DomainParticipant, so authentication is done at DomainParticipant level (not of single Publisher/Subscriber, not of single DataWriter/DataReader). The symmetric key to be used for a specific DomainParticipant shall be managed directly by CSM, which should provide a handle to DDS to use its services.

For the above mentioned purposes, the DDS **Cryptographic Plugin** is used, which offers an interface to protect the whole RTPS message. The resulting RTPS message, after security is applied, is shown in the picture below.

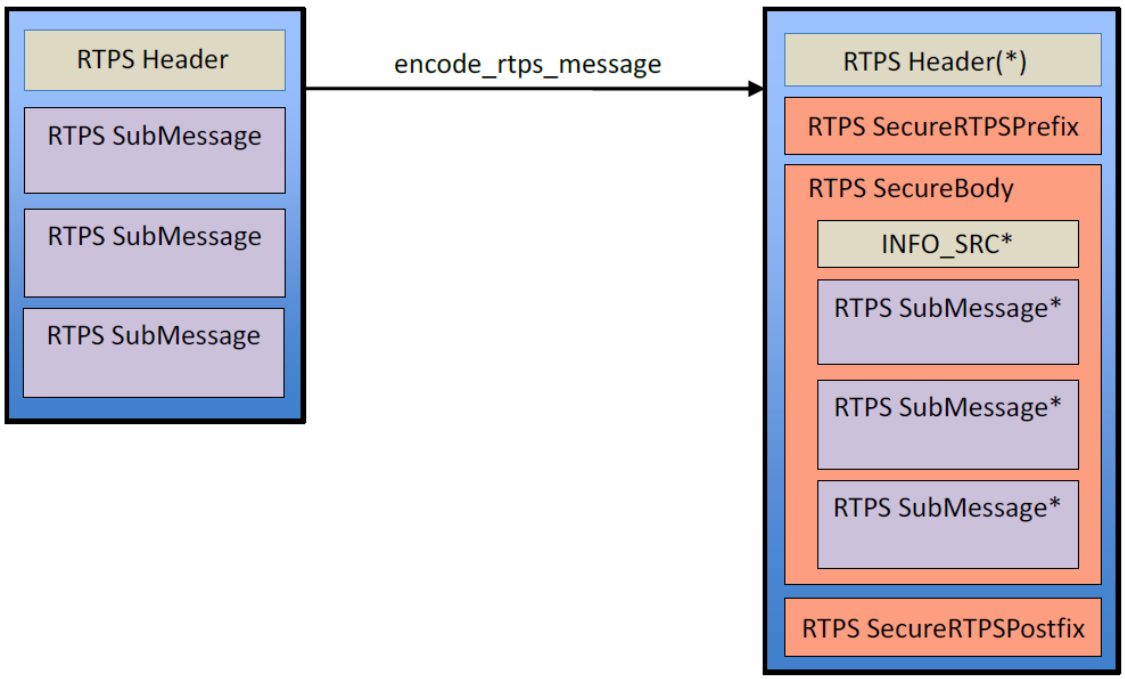

Figure 7.2

7.1.3 Safety Mechanisms

According to ISO 26262, there is a set of faults that can be considered on communication link between senders and receivers executed in different software partitions or ECUs.

The concept of end-to-end protection assumes that safety-related data exchange shall be protected at runtime against the effects of faults within the communication link.

The DDS Specification has intrinsic safety mechanisms(Counters, CRCs, QoS policies) that can be used to support a safety argument.

The following is a list of possible faults, as defined in [12], to be addressed in the pursuit of functional safety, and the mechanisms DDS offers to support them:

- **Repetition, loss, insertion, incorrect sequence, information from a sender received by only a subset of receivers, and blocking access to a communication channel:** submessage 64-bit sequence number, as defined in [6] section 8 3.5.4 "SequenceNumber", and additional SequenceNumber-typed fields in section 8.3.7 "RTPS Submessages"
- **Delay of information and blocking access to a communication channel:** LATENCY_BUDGET Quality of Service policy, as defined in [1] section 2.2.3.8 "LATENCY_BUDGET"
- **Masquerade or incorrect addressing of information:** DDS Security authentication plugin, as defined in [11] section 8.3 "Authentication Plugin". In this concept, only Autentication at DomainParticipant level could be achieved, since all entities belonging to the same DomainParticipant level share the same symmetric key. This prevent entities external to DomainParticipant to have access to the DomainParticipant communication, but it does prevent to distinguish 2 different entities that are authorized to communicate inside the DomainParticipant.
- **Corruption of information, asymmetric information sent from a sender to multiple receivers:** rtpsMessageChecksum under HeaderExtension submessage ([RTPS 2.5 or higher]). In absence of this feature, [11] also provides message integrity verification built into its message authentication protocol. For CRC calculation, the AUTOSAR CRC library is used.
- **Translation of these fault conditions errors:** In case of any communication errors or fault, the Dds BSW should notify the Det module.

7.2 General Requirements

[CP_SWS_Dds_CONSTR_00712]{DRAFT} **Topic name uniqueness** ⌈The Dds BSW module shall reject a configuration where different Topics belonging to the same DomainParticipant share the same name.⌋ *(FO_RS_Dds_00008)*

7.2.1 Communication requirements

[CP_SWS_Dds_CONSTR_00725]{DRAFT} **No data serialization** ⌈The Dds Bsw Module shall receive data as produced by applications into a DdsUpperLayerIPdu. No data serialization or data transformation shall be performed before the Dds BSW Module is invoked. (refer to `DdsDataWriter`, field DdsUpperLayerIPduRef).

During validation of configuration, the RTE configuration shall be checked. If any ISignal belonging to Dds BSW is linked to a Transformer, the validation shall return an error.⌋*(FO_RS_Dds_00001, FO_RS_Dds_00004, FO_RS_Dds_CONSTR_00006)*

[CP_SWS_Dds_00734]{DRAFT} **DDS Data serialization** ⌈The Dds BSW shall perform the actual serialization of data into a DdsLowerLayerIPdu (refer to `DdsDataWriter`, field DdsLowerLayerIPduRef). On receiving side, the Dds BSW shall deserialize data to a DdsUpperLayerIPdu (refer to `DdsDataReader`, field DdsUpperLayerIPdu) .⌋*(FO_RS_Dds_00001, FO_RS_Dds_00004, FO_RS_Dds_00019, FO_RS_Dds_00020)*

[CP_SWS_Dds_00726]{DRAFT} **RTPS compliance** ⌈Data produced into the DdsLowerLayerIPdu shall be compliant with the DDS Wire Interoperability protocol (RTPS) defined in [6].⌋*(FO_RS_Dds_00002, FO_RS_Dds_00017, FO_RS_Dds_00019, FO_RS_Dds_00020)*

[CP_SWS_Dds_00727]{DRAFT} **DDS standard serialization/deserialization rules** ⌈The Dds BSW module shall serialize the payload according to the DDS standard serialization rules defined in section 7.4.3.5 of [13].⌋*(FO_RS_Dds_00004, FO_RS_Dds_00007, FO_RS_Dds_00019, FO_RS_Dds_00020)*

[CP_SWS_Dds_00728]{DRAFT} **DDS serialization of primitive types** ⌈The Dds BSW module shall serialize AUTOSAR primitive data types according to the standard serialization rules for the equivalent DDS PRIMITIVE_TYPE defined in section 7.2.2.2 of [13]⌋*(FO_RS_Dds_00004, FO_RS_Dds_00007)*

Table 7.1 provides the equivalent DDS PRIMITIVE_TYPEs for the primitive AUTOSAR data type . For AUTOSAR CP data types, please refer to [14], chapter 8.2

Type	DDS Type	Remark
boolean	Boolean	
uint8	Byte	
uint16	UInt16	
uint32	UInt32	
uint64	UInt64	
sint8	Byte	
sint16	Int16	
sint32	Int32	
sint64	Int64	
uint8_least	Byte	
uint16_least	UInt16	
uint32_least	UInt32	
sint8_least	Byte	
sint16_least	Int16	

sint32_least	Int32	
float32	Float32	
float64	Float64	

Table 7.1: Serialization of primitive data types

[CP_SWS_Dds_00729]{DRAFT} **DDS serialization of enumeration data types** ⌈The Dds BSW module shall serialize ImplementationDataType of category ENUMERATION (refer to chapter 5.5.4 of [4]) according to the standard serialization rules for DDS ENUM_TYPE defined in section 7.2.2.4.1.1 of [13].⌋*(FO_RS_Dds_00004, FO_RS_Dds_00007)*

[CP_SWS_Dds_00730]{DRAFT} **DDS serialization of ARRAY data type** ⌈The Dds BSW module shall serialize ImplementationDataType of category ARRAY (refer to chapter 5.3.4.4 of [4]) according to the standard serialization rules for DDS ARRAY_TYPE defined in section 7.2.2.4.3 of [13].⌋*(FO_RS_Dds_00004, FO_RS_Dds_00007)*

[CP_SWS_Dds_00731]{DRAFT} **DDS serialization of STRUCTURE data type** ⌈The Dds BSW module shall serialize ImplementationDataType of category STRUCTURE (refer to chapter 7.2.2.4.4.1 of [4]) according to the standard serialization rules for DDS STRUCT_TYPE defined in section 7.4.3.5 of [13]. The Dds BSW module shall mark as optional all optional members of the structure (refer to section 7.2.2.4.4.5 of [13].)⌋ *(FO_RS_Dds_00004, FO_RS_Dds_00007)*

[CP_SWS_Dds_CONSTR_00732]{DRAFT} **DDS serialization of UNION data type** ⌈ImplementationDataType of category UNION (refer to chapter 7.2.2.4.4.2 of [4]) are not managed by the Dds BSW.

The Dds BSW configuration validation shall fail in case a DdsTopic links an ImplementationDataType which contains an union.⌋*(FO_RS_Dds_00004, FO_RS_Dds_00007)*

[CP_SWS_Dds_CONSTR_00733]{DRAFT} **DDS serialization of POINTER data type** ⌈ImplementationDataType of category POINTER (refer to chapter 7.2.2.4.6 of [4]) are not managed by the Dds BSW.

The Dds BSW configuration validation shall fail in case a DdsTopic links an ImplementationDataType which contains a pointer.⌋*(FO_RS_Dds_00004, FO_RS_Dds_00007)*

[CP_SWS_Dds_00735]{DRAFT} **Encoding Format and Endianness of Strings in DDS** ⌈The Dds BSW module shall encode Strings according Section 7.4.1.1.2 of [13].⌋ *(FO_RS_Dds_00004, FO_RS_Dds_00007)*

7.2.2 Security requirements

[CP_SWS_Dds_00750]{DRAFT} **DDS-security** ⌈In order to be compliant and to intercommunicate with other DDS systems, the Dds BSW module shall implement security mechanisms by using DDS-Security Specification [11].⌋*(FO_RS_Dds_00009)*

[CP_SWS_Dds_00752]{DRAFT} **MAC usage** ⌈The Dds BSW module shall guarantee data-integrity and message authentication at DomainParticipant level by adding a Message Authentication Code (MAC) to the message to be sent, calculated by using symmetric key algorithms. The resulting message shall still be DDSI-RTPS compliant.⌋ *(FO_RS_Dds_00009)*

[CP_SWS_Dds_00753]{DRAFT} **CSM library usage** ⌈The Dds BSW shall configure, for each DomainParticipant, one reference to each CSM job needed: one job to calculate MAC (DdsDomainParticipantCsmAuthenticateJob) and one to check MAC of received messages (DdsDomainParticipantCsmVerifyJob). For configuration details, refer to `DdsCryptoInfo`. At sender side, the Dds BSW shall add the resulting MAC of DdsDomainParticipantCsmAuthenticateJob to each message of this DomainParticipant. At receiving side, the Dds BSW shall check the result of the DdsDomainParticipantCsmVerifyJob.⌋ *(FO_RS_Dds_00009)*

[CP_SWS_Dds_CONSTR_00754]{DRAFT} **CSM job configuration** ⌈The CSM DdsDomainParticipantCsmAuthenticateJob shall be configured to call CSM_MacGenerate and the DdsDomainParticipantCsmVerifyJob to call CSM_MacVerify.

The Dds BSW configuration validation shall fail in case the DdsDomainParticipantCsmAuthenticateJob/DdsDomainParticipantCsmVerifyJob related to the same DdsDomainParticipant link CSM jobs that are not configured with CSM_MacGenerate and CSM_MacVerify respectly.

For configuration details, refer to `DdsCryptoInfo`.⌋ *(FO_RS_Dds_00009)*

[CP_SWS_Dds_CONSTR_00743]{DRAFT} **CSM key configuration** ⌈Each CSM authenticate/verify pair, related to a single DomainParticipant, shall use the same keys (only simmetric-key algorithms are supported). For each DomainParticipant, the CSM used jobs shall be configured with the same keys.

The Dds BSW configuration validation shall fail in case the DdsDomainParticipantCsmAuthenticateJob/DdsDomainParticipantCsmVerifyJob related to the same DdsDomainParticipant link CSM jobs that are not configured with the same key.

For configuration details, refer to `DdsCryptoInfo`.⌋ *(FO_RS_Dds_00009)*

[CP_SWS_Dds_00756]{DRAFT} **MAC calculation failure** ⌈If the MAC calculation fails (e.g. the Csm_MacGenerate() or Csm_MacVerify() return any error), the Dds BSW module shall call the API Det_ReportRuntimeError() with the DDS_E_CSM_LIBRARY_ERROR runtime error code and discard the message to be sent.

In this case, during transmission the Dds BSW shall call the PduR_DdsTxConfirmation function with result = E_NOT_OK.⌋ *(FO_RS_Dds_00009)*

[CP_SWS_Dds_00758]{DRAFT} **MAC check failure** ⌈At receiving side, if the MAC check fails, the Dds BSW module shall call the API Det_ReportRuntimeError() with the DDS_E_CSM_CHECK_FAILED rutime error code and discard the message.⌋ *(FO_RS_Dds_00009)*

7.2.3 Safety requirements

[CP_SWS_Dds_00761]{DRAFT} **Repetition or Insertion of Information** ⌈The Dds BSW module shall use submessages which have counters, e.g AckNack, Data and DataFrag, etc., to guarantee safety mechanisms against Repetition or Insertion of Information faults. At receiving side, if a message with a duplicated counter is received, the Dds BSW module shall discard the message and call the API Det_ReportRuntimeError() with the DDS_E_SAMPLE_REJECTED runtime error code.⌋*(FO_RS_Dds_00010)*

[CP_SWS_Dds_00762]{DRAFT} **Loss or Incorrect sequence of Information** ⌈The Dds BSW module shall use submessages which have counters, e.g AckNack, Data and DataFrag, etc., to guarantee safety mechanisms against Loss or Incorrect sequence of Information faults. At receiving side, if a message with a non-consecutive counter is received, the Dds BSW module shall discard the message and call the API Det_ReportRuntimeError() with the DDS_E_SAMPLE_LOST runtime error code.⌋ *(FO_RS_Dds_00010)*

[CP_SWS_Dds_00763]{DRAFT} **Delay of Information** ⌈The Dds BSW module shall use QoSs able to monitor timeouts, such as DEADLINE, LATENCY_BUDGET, LIFESPAN and TIME_BASED_FILTER (refer respectively to `DdsDeadline`, `DdsLatencyBudget`, `DdsLifespan` and `DdsTimeBasedFilter`) to guarantee safety mechanisms against Delay of Information fault (take [1] for details on those QoS policies). At sending side, if some timing contraint is not fullfilled, the Dds BSW module shall discard the message and call the API Det_ReportRuntimeError() with the DDS_E_SENDER_TIMING_MISSED runtime error code. At receiving side, if some timing contraint is not fullfilled, the Dds BSW module shall discard the message and call the API Det_ReportRuntimeError() with the DDS_E_RECEIVER_TIMING_MISSED runtime error code.⌋*(FO_RS_Dds_00005, FO_RS_Dds_00010)*

[CP_SWS_Dds_00766]{DRAFT} **Corruption of Information** ⌈The Dds BSW module use CRC check to guarantee safety mechanisms against Corruption of Information fault.⌋*(FO_RS_Dds_00010)*

[CP_SWS_Dds_00769]{DRAFT} **CRC check failure** ⌈On received side, if the CRC check fails, the Dds BSW module shall call the API Det_ReportRuntimeError() with the DDS_E_CRC_CHECK_FAILED runtime error code and discard the message.⌋*(FO_RS_Dds_00010)*

7.3 Error Classification

Section 7.2 "Error Handling" of the document "General Specification of Basic Software Modules" [3] describes the error handling of the Basic Software in detail. Above all, it constitutes a classification scheme consisting of five error types which may occur in BSW modules.

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

7.3.1 Development Errors

[CP_SWS_Dds_00772]{DRAFT} ⌈

Type of error	*Related error code*	*Error value*
Module not initialized	DDS_E_UNINIT	0x00
Null pointer has been passed as an argument	DDS_E_PARAM_POINTER	0x02
Invalid Upper Layer PduId	DDS_E_U_PDUID_INVALID	0x03
Invalid Lower Layer PduId	DDS_E_L_PDUID_INVALID	0x04

⌋()

7.3.2 Runtime Errors

[CP_SWS_Dds_00773]{DRAFT} ⌈

Type of error	*Related error code*	*Error value*
Upper layer module request rejected	DDS_E_U_PDUID_REJECTED	0x10
Lower layer notify ignored	DDS_E_L_PDUID_IGNORED	0x11
CSM library error	DDS_E_CSM_LIBRARY_ERROR	0x30
CSM check error	DDS_E_CSM_CHECK_FAILED	0x40
CRC check failed	DDS_E_CRC_CHECK_FAILED	0x41
Sample rejected	DDS_E_SAMPLE_REJECTED	0x42
Sample lost	DDS_E_SAMPLE_LOST	0x43
Timing constraints missed at receiver side	DDS_E_RECEIVER_TIMING_MISSED	0x44
Timing constraints missed at sender side	DDS_E_SENDER_TIMING_MISSED	0x45
Internal error	DDS_INTERNAL_ERROR	0x46

⌋()

7.3.3 Production Errors

There are no production errors.

7.3.4 Extended Production Errors

There are no extended production errors.

8 API specification

8.1 Imported types

In this chapter all types included from the following files are listed.

[CP_SWS_Dds_00801]{DRAFT} ⌈

Module	*Header File*	*Imported Type*
ComStack_Types	ComStack_Types.h	PduIdType
	ComStack_Types.h	PduInfoType
	ComStack_Types.h	PduLengthType
Csm	Rte_Csm_Type.h	Crypto_OperationModeType
	Rte_Csm_Type.h	Crypto_VerifyResultType
StbM	Rte_StbM_Type.h	StbM_SynchronizedTimeBaseType
	Rte_StbM_Type.h	StbM_TimeBaseStatusType
	Rte_StbM_Type.h	StbM_TimeStampType
	Rte_StbM_Type.h	StbM_TimeTupleType
	Rte_StbM_Type.h	StbM_UserDataType
	StbM.h	StbM_VirtualLocalTimeType
Std	Std_Types.h	Std_ReturnType
	Std_Types.h	Std_VersionInfoType

⌋()

8.2 Type definitions

8.2.1 Dds_ConfigType

[CP_SWS_Dds_00802]{DRAFT} ⌈

Name	Dds_ConfigType (draft)	
Kind	Structure	
Elements	implementation specific	
	Type	–
	Comment	The content of the initialization data structure is implementation specific
Description	This is the type of the data structure containing the initialization data for Dds. **Tags:** atp.Status=draft	
Available via	Dds.h	

⌋()

8.3 Function definitions

This is a list of functions provided for upper layer modules.

8.3.1 Dds_Init

[CP_SWS_Dds_00810]{DRAFT} ⌈

Service Name	Dds_Init (draft)	
Syntax	`void Dds_Init ( const Dds_ConfigType* Dds_ConfigPtr )`	
Service ID [hex]	0x00	
Sync/Async	Synchronous	
Reentrancy	Non Reentrant	
Parameters (in)	Dds_ConfigPtr	Pointer to a selected configuration structure
Parameters (inout)	None	
Parameters (out)	None	
Return value	None	
Description	This service initializes interfaces and variables of the AUTOSAR Dds layer. **Tags:** atp.Status=draft	
Available via	Dds.h	

⌋(*SRS_BSW_00405*, *SRS_BSW_00101*, *SRS_BSW_00414*)

[CP_SWS_Dds_00811]{DRAFT} **Dds_Init behaviour** ⌈The function Dds_Init shall initialize all module-related variables and constants according configuration.⌋(*SRS_BSW_00101*)

[CP_SWS_Dds_00812]{DRAFT} **Dds_Init - Entity state** ⌈The function Dds_Init shall initialize all Entities to "enabled" state.⌋(*SRS_BSW_00101*)

[CP_SWS_Dds_00813]{DRAFT} **Dds_Init - Buffer state** ⌈The function Dds_Init shall empty all internal buffers.⌋(*SRS_BSW_00101*)

8.3.2 Dds_GetVersionInfo

[CP_SWS_Dds_00820]{DRAFT} ⌈

Service Name	Dds_GetVersionInfo (draft)
Syntax	`void Dds_GetVersionInfo ( Std_VersionInfoType* versioninfo )`
Service ID [hex]	0x01
Sync/Async	Synchronous
Reentrancy	Reentrant

▽

△

Parameters (in)	None	
Parameters (inout)	None	
Parameters (out)	versioninfo	Pointer to where to store the version information of this module.
Return value	None	
Description	Returns the version information of this module. **Tags:** atp.Status=draft	
Available via	Dds.h	

⌋(*SRS_BSW_00402*, *SRS_BSW_00407*, *SRS_BSW_00411*, *SRS_BSW_00374*, *SRS_BSW_00379*, *SRS_BSW_00003*, *SRS_BSW_00318*)

[CP_SWS_Dds_00821]{DRAFT} **Dds_GetVersion - Null VersionInfoPtr** ⌈If development error detection for the Dds module is enabled, then the function Dds_GetVersionInfo shall check whether the parameter VersioninfoPtr is a NULL pointer (NULL_PTR). If VersioninfoPtr is a NULL pointer, then the function Dds_GetVersionInfo shall raise the development error DDS_E_PARAM_POINTER.⌋ *(SRS_BSW_00003)*

8.3.3 Dds_Transmit

The Dds module provides the Dds_Transmit function so that the PDU Router is able to initiate the transmission of a upper layer I-PDU.

[CP_SWS_Dds_00831]{DRAFT} ⌈

Service Name	Dds_Transmit (draft)	

Syntax	`Std_ReturnType Dds_Transmit (`
	`  PduIdType TxPduId,`
	`  const PduInfoType* PduInfoPtr`
	`)`	

Service ID [hex]	0x02	
Sync/Async	Synchronous	
Reentrancy	Reentrant for different PduIds. Non reentrant for the same PduId.	
Parameters (in)	TxPduId	Identifier of the PDU to be transmitted
	PduInfoPtr	Length of and pointer to the PDU data and pointer to MetaData.
Parameters (inout)	None	
Parameters (out)	None	
Return value	Std_ReturnType	E_OK: Transmit request has been accepted. E_NOT_OK: Transmit request has not been accepted.
Description	Request for transmitting a message. **Tags:** atp.Status=draft	
Available via	Dds.h	

⌋(*SRS_BSW_00402*, *SRS_BSW_00407*, *SRS_BSW_00411*, *SRS_BSW_00374*, *SRS_BSW_00379*, *SRS_BSW_00003*, *SRS_BSW_00318*)

[CP_SWS_Dds_00851]{DRAFT} **Internal transmission buffer presence** ⌈The Dds module shall manage an internal buffer where to save incoming upper layer I-PDUs. The size of the buffer shall be configurable.⌋*(FO_RS_Dds_00015)*

[CP_SWS_Dds_00852]{DRAFT} **Dds_Transmit - Error conditions** ⌈The function Dds_Transmit shall call the Det_ReportError, if development error detection is enabled and if function call has failed because of the following reasons:

- Dds module is not initialized (DDS_E_UNINIT)
- PduInfoPtr equals NULL_PTR (DDS_E_PARAM_POINTER).
- Invalid upper layer TxPduId (DDS_E_U_PDUID_INVALID).

  Besides the function Dds_Transmit shall call the API PduR_DdsTxConfirmation with result = E_NOT_OK.

⌋*(FO_RS_Dds_00015)*

[CP_SWS_Dds_00854]{DRAFT} **Dds_Transmit - DDS_E_U_PDUID_REJECTED** ⌈If upper layer module request cannot be accepted or the internal buffer is full, Dds_Transmit shall call the API Det_ReportRuntimeError with the runtime error code DDS_E_U_PDUID_REJECTED and then return E_NOT_OK.

Besides the function Dds_Transmit shall call the API PduR_DdsTxConfirmation with result = E_NOT_OK.⌋*(FO_RS_Dds_00015)*

[CP_SWS_Dds_00855]{DRAFT} **Dds_Transmit - E_OK** ⌈If upper layer module request can be accepted, Dds_Transmit shall store the upper layer I-PDU into the internal buffer, update the buffer offset which indicates where to save new I-PDUs and return E_OK.⌋*(FO_RS_Dds_00015)*

8.4 Callback notifications

This is a list of functions provided for other modules.

8.4.1 Dds_RxIndication

Every lower layer I-PDU which is received by the Ethernet Interface is given to the PDU Router by means of the SoAd. The PDU Router routes those I-PDU to the Dds reception interface invoking the Dds_RxIndication callback.

[CP_SWS_Dds_00841]{DRAFT} ⌈

Service Name	Dds_RxIndication (draft)	

Syntax	`void Dds_RxIndication (`
	`  PduIdType RxPduId,`
	`  const PduInfoType* PduInfoPtr`
	`)`	

Service ID [hex]	0x42	
Sync/Async	Synchronous	
Reentrancy	Reentrant for different PduIds. Non reentrant for the same PduId.	
Parameters (in)	RxPduId	ID of the received PDU.
	PduInfoPtr	Contains the length (SduLength) of the received PDU, a pointer to a buffer (SduDataPtr) containing the PDU, and the MetaData related to this PDU.
Parameters (inout)	None	
Parameters (out)	None	
Return value	None	
Description	Indication of a received PDU from the PDU Router interface module. **Tags:** atp.Status=draft	
Available via	Dds.h	

⌋*()*

[CP_SWS_Dds_00864]{DRAFT} **Internal reception buffer presence** ⌈The Dds module shall manage an internal buffer where to save incoming lower layer I-PDUs. The size of the buffer shall be configurable.⌋*(FO_RS_Dds_00016)*

[CP_SWS_Dds_00861]{DRAFT} **Dds_RxIndication - Error conditions** ⌈The function Dds_RxIndication shall call the Det_ReportError, if development error detection is enabled and if function call has failed because of the following reasons:

- Dds module is not initialized (DDS_E_UNINIT)
- PduInfoPtr equals NULL_PTR (DDS_E_PARAM_POINTER).
- Invalid lower layer RxPduId (DDS_E_L_PDUID_INVALID).

⌋*(FO_RS_Dds_00016)*

[CP_SWS_Dds_00862]{DRAFT} **Dds_RxIndication - DDS_E_L_PDUID_IGNORED** ⌈If lower layer module I-PDU reception cannot be accepted or the internal buffer is full, Dds_RxIndication shall call the API Det_ReportRuntimeError with the runtime error code DDS_E_L_PDUID_IGNORED and return.⌋*(FO_RS_Dds_00016)*

[CP_SWS_Dds_00863]{DRAFT} **Dds_RxIndication - OK condition** ⌈If lower layer module I-PDU reception can be accepted, the Dds_RxIndication shall store the lower layer I-PDU into the internal buffer and return.⌋*(FO_RS_Dds_00016)*

8.4.2 Dds_TxConfirmation

[CP_SWS_Dds_00843]{DRAFT} ⌈

Service Name	Dds_TxConfirmation (draft)	
Syntax	`void Dds_TxConfirmation ( PduIdType TxPduId, Std_ReturnType result )`	
Service ID [hex]	0x40	
Sync/Async	Synchronous	
Reentrancy	Reentrant for different PduIds. Non reentrant for the same PduId.	
Parameters (in)	TxPduId	ID of the PDU that has been transmitted.
	result	E_OK: The PDU was transmitted. E_NOT_OK: Transmission of the PDU failed.
Parameters (inout)	None	
Parameters (out)	None	
Return value	None	
Description	The PDU Router interface module confirms the transmission of a PDU, or the failure to transmit a PDU. **Tags:** atp.Status=draft	
Available via	Dds.h	

⌋*()*

[CP_SWS_Dds_00871]{DRAFT} **Dds_TxConfirmation - Error conditions** ⌈The function Dds_TxConfirmation() shall call the Det_ReportError(), if development error detection is enabled and if function call has failed because of the following reasons:

- Dds module is not initialized (DDS_E_UNINIT)
- Invalid lower layer TxPduId (DDS_E_L_PDUID_INVALID).

⌋*(FO_RS_Dds_00015)*

[CP_SWS_Dds_00872]{DRAFT} **Dds_TxConfirmation behaviour** ⌈Dds_TxConfirmation shall call the API PduR_DdsTxConfirmation with the same input parameters and return.⌋*(FO_RS_Dds_00015)*

8.4.3 Dds_TriggerTransmit

[CP_SWS_Dds_00835]{DRAFT} ⌈

Service Name	Dds_TriggerTransmit (draft)
Syntax	`Std_ReturnType Dds_TriggerTransmit ( PduIdType TxPduId, PduInfoType* PduInfoPtr )`
Service ID [hex]	0x41
Sync/Async	Synchronous

▽

△

Reentrancy	Reentrant for different PduIds. Non reentrant for the same PduId.	
Parameters (in)	TxPduId	ID of the SDU that is requested to be transmitted.
Parameters (inout)	PduInfoPtr	Contains a pointer to a buffer (SduDataPtr) to where the SDU data shall be copied, and the available buffer size in SduLengh. On return, the service will indicate the length of the copied SDU data in SduLength.
Parameters (out)	None	
Return value	Std_ReturnType	E_OK: SDU has been copied and SduLength indicates the number of copied bytes. E_NOT_OK: No SDU data has been copied. PduInfoPtr must not be used since it may contain a NULL pointer or point to invalid data.
Description	Within this API, the upper layer module (called module) shall check whether the available data fits into the buffer size reported by PduInfoPtr->SduLength. If it fits, it shall copy its data into the buffer provided by PduInfoPtr->SduDataPtr and update the length of the actual copied data in PduInfoPtr->SduLength. If not, it returns E_NOT_OK without changing PduInfoPtr. **Tags:** atp.Status=draft	
Available via	Dds.h	

⌋*()*

[CP_SWS_Dds_00881]{DRAFT} **Dds_TriggerTransmit - Error conditions** ⌈The function Dds_TriggerTransmit() shall call the Det_ReportError(), if development error detection is enabled and if function call has failed because of the following reasons:

- Dds module is not initialized (DDS_E_UNINIT)
- PduInfoPtr equals NULL_PTR (DDS_E_PARAM_POINTER).
- Invalid lower layer TxPduId (DDS_E_L_PDUID_INVALID).

⌋*(FO_RS_Dds_00015)*

[CP_SWS_Dds_00882]{DRAFT} **Dds_TriggerTransmit behaviour** ⌈Within the function Dds_TriggerTransmit(), the Dds BSW shall copy the contents of its I-PDU transmit buffer to the PDU buffer given by PduInfoPtr->SduDataPtr and update PduInfoPtr->SduLength with length of the copied data accordingly.⌋*(FO_RS_Dds_00015)*

[CP_SWS_Dds_00883]{DRAFT} **Dds_TriggerTransmit - Error conditions** ⌈If another Dds_TriggerTransmit() request on the same PduId is not yet completed, the function Dds_TriggerTransmit() shall call the Det_ReportRuntimeError() with the code DDS_E_L_PDUID_IGNORED.⌋*(FO_RS_Dds_00015)*

[CP_SWS_Dds_00884]{DRAFT} **Dds_TriggerTransmit limitation** ⌈The Dds_TriggerTransmit function can be called only by lower-layer module. During validation of configuration, the LdCom configuration shall be checked. If any I-PDU belonging to Dds BSW has a non NULL value for LdComTxTriggerTrasmit, the validation shall return an error.⌋*(FO_RS_Dds_00015)*

8.5 Scheduled functions

Following functions are called directly by Basic Software Scheduler. They have no return value and no parameter. All functions shall be non-reentrant

8.5.1 Dds_RxMainFunction

[CP_SWS_Dds_00823]{DRAFT} ⌈

Service Name	Dds_RxMainFunction (draft)
Syntax	`void Dds_RxMainFunction ( void )`
Service ID [hex]	0x10
Description	Scheduled function of the Dds module for reception purpose **Tags:** atp.Status=draft
Available via	SchM_Dds.h

⌋*(SRS_BSW_00424, SRS_BSW_00433, SRS_BSW_00373)*

[CP_SWS_Dds_00825]{DRAFT} **Dds_RxMainFunction** ⌈The Dds_RxMainFunction() should check if any PDU is present into internal reception buffer. If so, the Dds_RxMainFunction() shall perform all the DDS Middleware stuff (deserialization, QoS managment, safety and security tasks) on received data.⌋*(FO_RS_Dds_00016)*

[CP_SWS_Dds_00826]{DRAFT} **Dds_RxMainFunction - Error conditions** ⌈If, during processing of the Dds_RxMainFunction() function any error or violation occurres, the Dds_RxMainFunction shall call the Det_ReportRuntimeError() with the DDS_INTERNAL_ERROR code, drop received data and return.⌋*(FO_RS_Dds_00016)*

[CP_SWS_Dds_00827]{DRAFT} **Dds_RxMainFunction - OK conditions** ⌈If, during processing of the Dds_RxMainFunction() function, everything is ok, the Dds_RxMainFunction shall find the proper readers to manage the reception of data and shall call the API PduR_SoAdRxIndication with the resulting upper layer I-PDU as input parameter.⌋*(FO_RS_Dds_00016)*

8.5.2 Dds_TxMainFunction

[CP_SWS_Dds_00824]{DRAFT} ⌈

Service Name	Dds_TxMainFunction (draft)
Syntax	`void Dds_TxMainFunction ( void )`
Service ID [hex]	0x11

▽

△

Description	Scheduled function of the Dds module for transmission purpose **Tags:** atp.Status=draft
Available via	SchM_Dds.h

⌋*(SRS_BSW_00424, SRS_BSW_00433, SRS_BSW_00373)*

[CP_SWS_Dds_00828]{DRAFT} **Dds_TxMainFunction** ⌈The Dds_TxMainFunction() should check if any PDU is present into internal transmission buffer. If so, the Dds_TxMainFunction() shall perform all the DDS Middleware stuff (QoS managment, safety and security tasks, serialization) on received data.⌋*(FO_RS_Dds_00016)*

[CP_SWS_Dds_00829]{DRAFT} **Dds_TxMainFunction - Error conditions** ⌈If, during processing of the Dds_TxMainFunction() function any error or violation occurres, the Dds_TxMainFunction shall call the Det_ReportRuntimeError() with the DDS_INTERNAL_ERROR code, shall call the PduR_DdsTxConfirmation with result = E_NOT_OK, shall drop received data and return.⌋*(FO_RS_Dds_00015)*

[CP_SWS_Dds_00830]{DRAFT} **Dds_TxMainFunction - OK conditions** ⌈If, during processin of the Dds_RxMainFunction() function, everything is ok, the Dds_TxMainFunction shall find the proper writers to manage the transmission of data and shall call the API PduR_DdsTransmit with the resulting lower layer I-PDU as input parameter.⌋*(FO_RS_Dds_00015)*

8.6 Expected interfaces

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

8.6.1 Mandatory interfaces

[CP_SWS_Dds_00832]{DRAFT} ⌈

API Function	***Header File***	***Description***
Det_ReportError	Det.h	Service to report development errors.
Det_ReportRuntimeError	Det.h	Service to report runtime errors. If a callout has been configured then this callout shall be called.
PduR_DdsRxIndication (draft)	PduR_Dds.h	Indication of a received PDU from a lower layer communication interface module.
PduR_DdsTransmit (draft)	PduR_Dds.h	Requests transmission of a PDU.
PduR_DdsTxConfirmation (draft)	PduR_Dds.h	The lower layer communication interface module confirms the transmission of a PDU, or the failure to transmit a PDU.

⌋*()*

8.6.2 Optional interfaces

[CP_SWS_Dds_00833]{DRAFT} ⌈

API Function	*Header File*	*Description*
Crc_CalculateCRC32	Crc.h	This service makes a CRC32 calculation on Crc_ Length data bytes.
Crc_CalculateCRC64	Crc.h	This service makes a CRC64 calculation on Crc_ Length data bytes, using the polynomial 0x42F0E1EBA9EA3693. This CRC routine is used by E2E Profile 7.
Csm_MacGenerate	Csm.h	Uses the given data to perform a MAC generation and stores the MAC in the memory location pointed to by the MAC pointer.
Csm_MacVerify	Csm.h	Verifies the given MAC by comparing if the MAC is generated with the given data.
StbM_GetCurrentTime	StbM.h	Returns a time value (Local Time Base derived from Global Time Base) in standard format. Note: This API shall be called with locked interrupts / within an Exclusive Area to prevent interruption (i.e., the risk that the time stamp is outdated on return of the function call).

⌋()

8.6.3 Configurable interfaces

None.

9 Sequence diagrams

9.1 Transmission

9.1.1 Dds message transmission

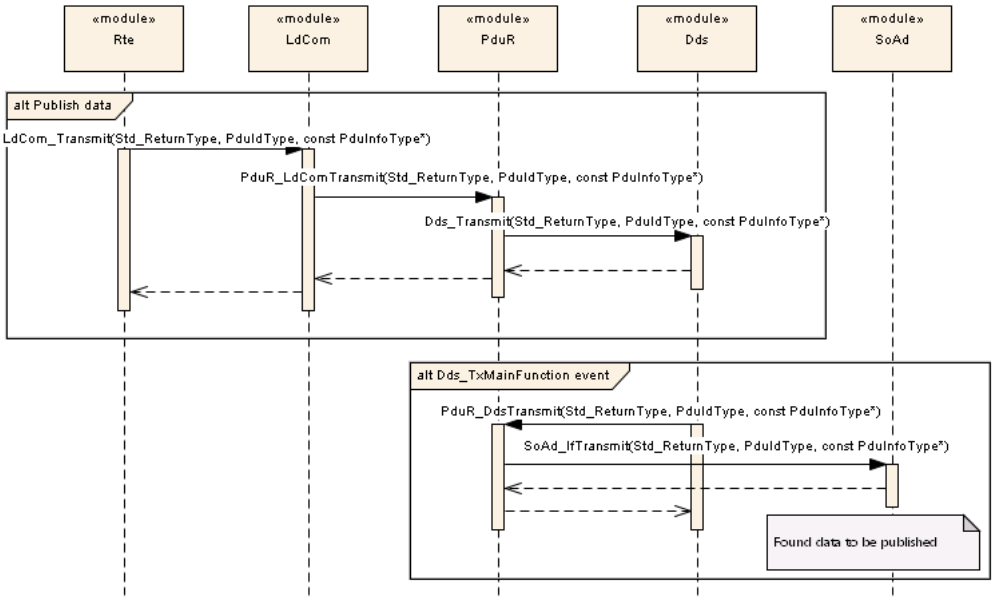

Figure 9.1: Dds transmission path

9.1.2 Dds message transmission confirmation

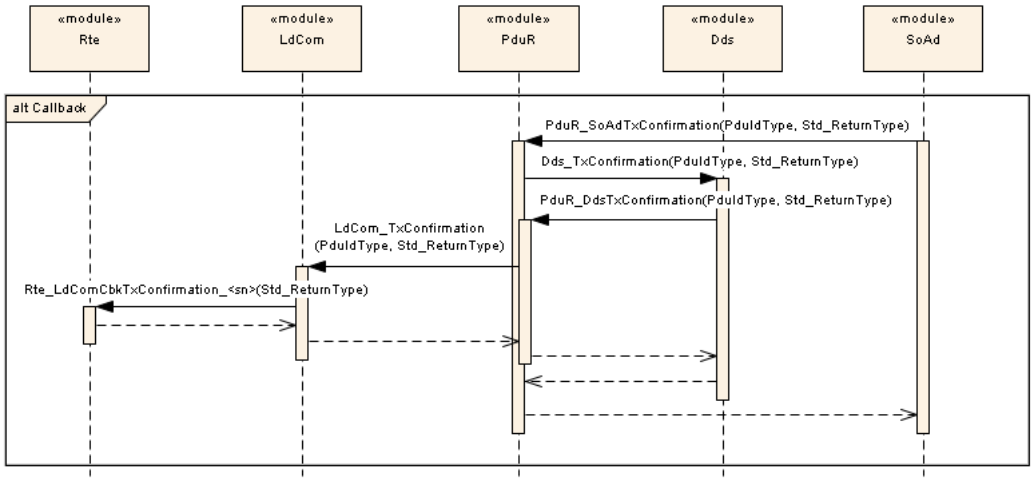

Figure 9.2: Dds transmission confirmation path

9.2 Reception

9.2.1 Dds received indication event

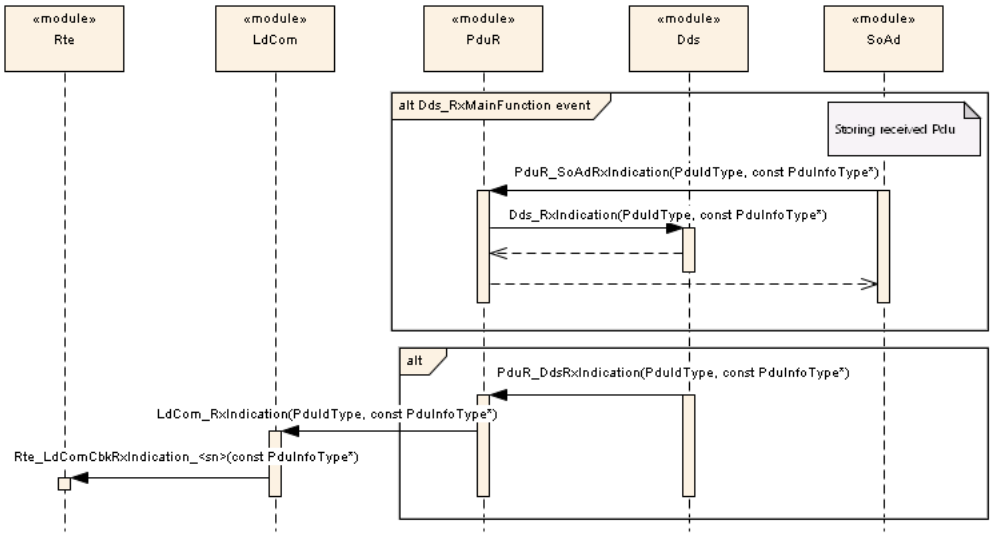

Figure 9.3: Dds reception path

10 Configuration specification

In general, this chapter defines configuration parameters and their clustering into containers. In order to support the specification Chapter 10.1 describes fundamentals. It also specifies a template (table) you shall use for the parameter specification. We intend to leave Chapter 10.1 in the specification to guarantee comprehension.

Chapter 10.2 specifies the structure (containers) and the parameters of the module Dds.

Chapter 10.3 specifies published information of the module Dds.

10.1 How to read this chapter

For details refer to the chapter 10.1 “Introduction to configuration specification” in SWS_BSWGeneral.

10.2 Containers and configuration parameters

The following chapters summarize all configuration parameters.

10.2.1 Dds

SWS Item	[ECUC_Dds_00001]
Module Name	Dds
Description	Configuration of the Dds module.
Post-Build Variant Support	true
Supported Config Variants	VARIANT-POST-BUILD, VARIANT-PRE-COMPILE

Included Containers		
Container Name	Multiplicity	Scope / Dependency
DdsConfig	1	This container contains the configuration parameters and sub containers of the AUTOSAR Dds module. **Tags:** atp.Status=draft
DdsGeneral	1	This container contains the general configuration parameters of the DDS. **Tags:** atp.Status=draft

In the picture below, the UML diagram of Dds BSW is shown:

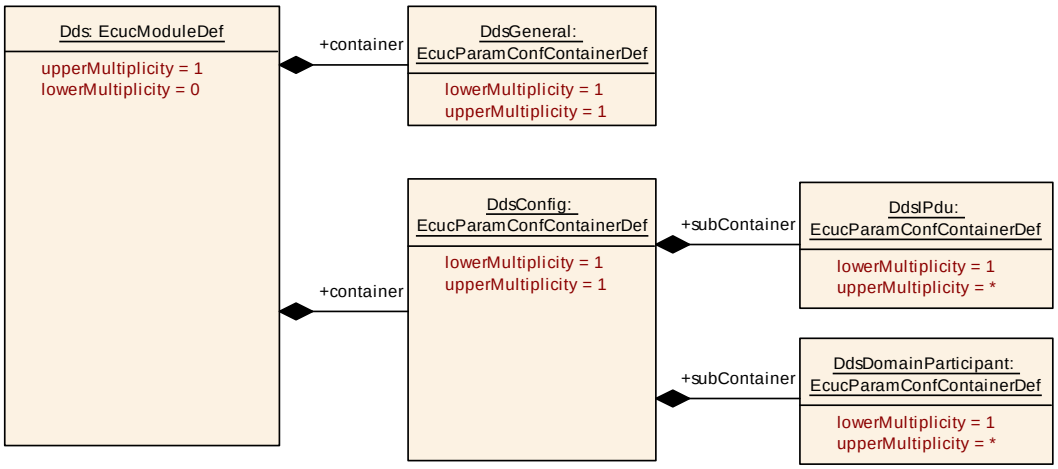

Figure 10.1: Dds model

10.2.2 Dds General

SWS Item	[ECUC_Dds_00002]		
Container Name	DdsGeneral		
Parent Container	Dds		
Description	This container contains the general configuration parameters of the DDS. **Tags:** atp.Status=draft		
Configuration Parameters			

SWS Item	[ECUC_Dds_00003]		
Parameter Name	DdsDevErrorDetect		
Parent Container	DdsGeneral		
Description	Switches the development error detection and notification on or off. • true: detection and notification is enabled. • false: detection and notification is disabled. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucBooleanParamDef		
Default value	false		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: local		

SWS Item	[ECUC_Dds_00130]
Parameter Name	DdsLowerLayerIPduBufferSize
Parent Container	DdsGeneral

▽

△

Description	This parameter defines the lower layer IPdu buffer size in byte. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucIntegerParamDef		
Range	0 .. 4294967295		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: local		

SWS Item	**[ECUC_Dds_00004]**		
Parameter Name	DdsMainRxFunctionPeriod		
Parent Container	DdsGeneral		
Description	This parameter defines the cycle time in seconds of the periodic call of the Dds_Rx MainFunction. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucFloatParamDef		
Range	]0 .. INF[		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: local		

SWS Item	**[ECUC_Dds_00127]**		
Parameter Name	DdsMainTxFunctionPeriod		
Parent Container	DdsGeneral		
Description	This parameter defines the cycle time in seconds of the periodic call of the Dds_Tx MainFunction. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucFloatParamDef		
Range	]0 .. INF[		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: local		

SWS Item	**[ECUC_Dds_00129]**
Parameter Name	DdsUpperLayerIPduBufferSize
Parent Container	DdsGeneral

▽

△

Description	This parameter defines the upper layer IPdu buffer size in byte. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucIntegerParamDef		
Range	0 .. 4294967295		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: local		

SWS Item	**[ECUC_Dds_00128]**		
Parameter Name	DdsSynchronizedTimeBaseRef		
Parent Container	DdsGeneral		
Description	Reference to a StbM Synchronized Time Base. **Tags:** atp.Status=draft		
Multiplicity	0..1		
Type	Symbolic name reference to StbMSynchronizedTimeBase		
Post-Build Variant Multiplicity	false		
Post-Build Variant Value	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

No Included Containers

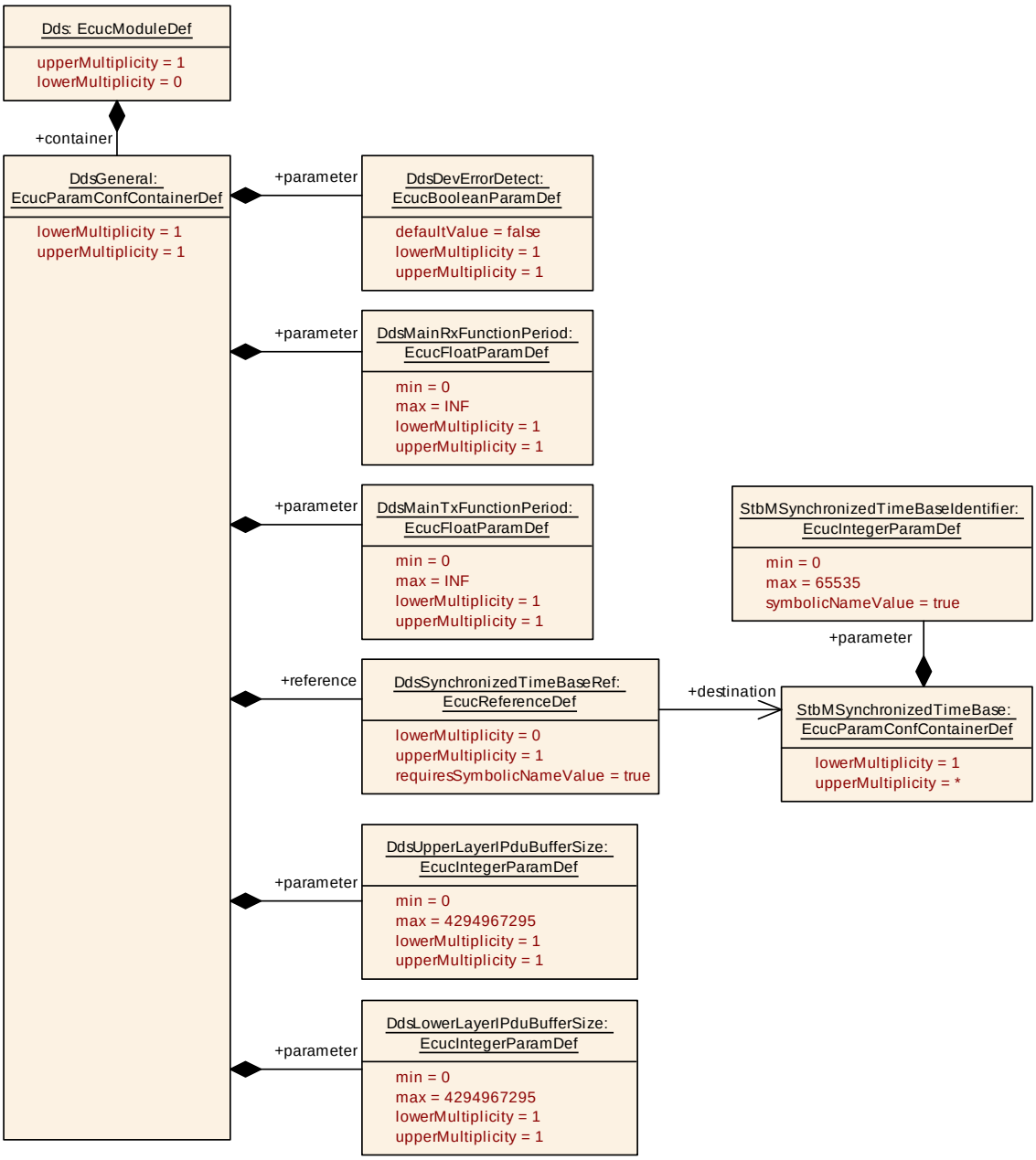

Figure 10.2: Dds General

10.2.3 Dds Config

SWS Item	[ECUC_Dds_00005]
Container Name	DdsConfig
Parent Container	Dds

▽

△

Description	This container contains the configuration parameters and sub containers of the AUTOSAR Dds module. **Tags:** atp.Status=draft
Configuration Parameters	

Included Containers		
Container Name	**Multiplicity**	**Scope / Dependency**
DdsDomainParticipant	1..*	This container represents the configuration of one single "Domain Participant" to which the current node belongs. One node can belong to more than one Domain Participant. One node can communicate only with other nodes beloning to the same Domain Participant. **Tags:** atp.Status=draft
DdsIPdu	1..*	This container contains the configuration of the IPdus used by the Dds module **Tags:** atp.Status=draft

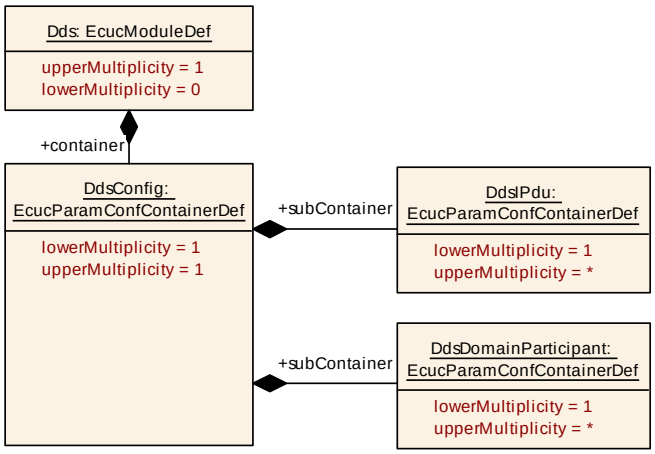

Figure 10.3: Dds Config

10.2.3.1 DdsIPdu

In the picture below, the UML diagram of DdsIPdu template is shown:

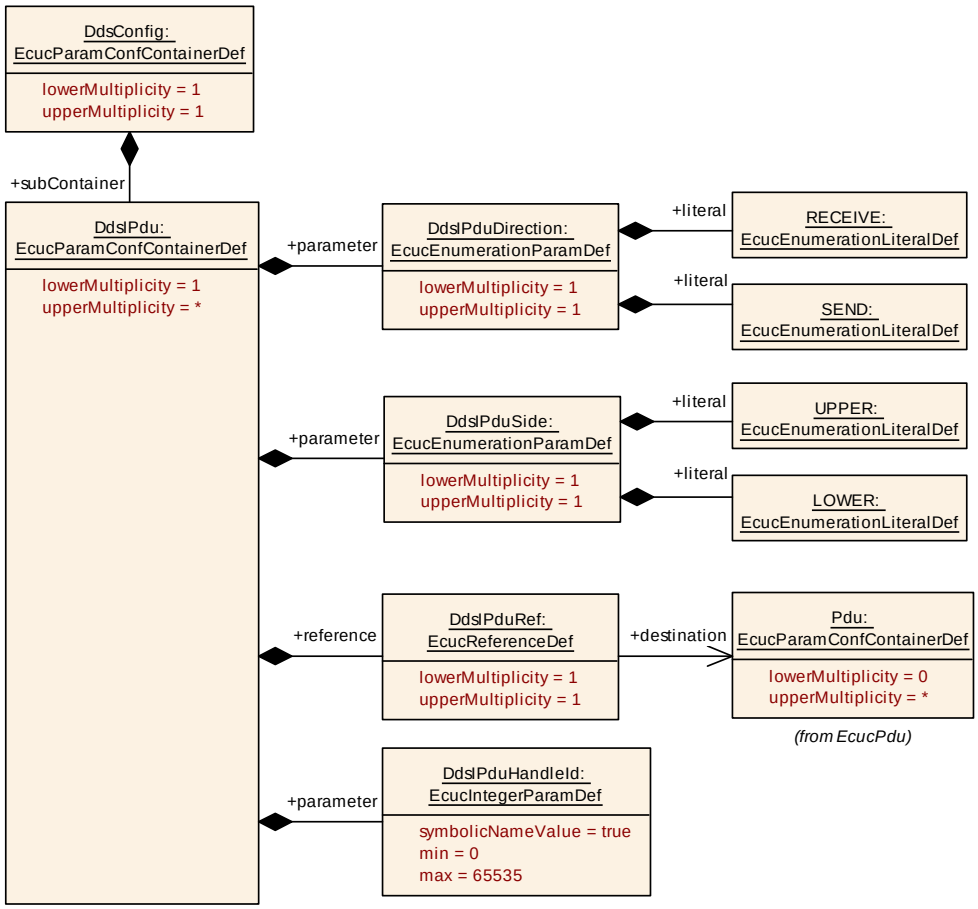

Figure 10.4: Dds IPdu

SWS Item	**[ECUC_Dds_00006]**		
Container Name	DdsIPdu		
Parent Container	DdsConfig		
Description	This container contains the configuration of the IPdus used by the Dds module **Tags:** atp.Status=draft		
Post-Build Variant Multiplicity	true		
Multiplicity Configuration Class	**Pre-compile time**	X	VARIANT-PRE-COMPILE
	Link time	–	
	Post-build time	X	VARIANT-POST-BUILD
Configuration Parameters			

SWS Item	**[ECUC_Dds_00008]**
Parameter Name	DdsIPduDirection
Parent Container	DdsIPdu
Description	Identify the direction of the IPdu. **Tags:** atp.Status=draft
Multiplicity	1
Type	EcucEnumerationParamDef

▽

△

Range	RECEIVE	IPdu for receiving. **Tags:** atp.Status=draft	
	SEND	IPdu for sending. **Tags:** atp.Status=draft	
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

SWS Item	**[ECUC_Dds_00011]**		
Parameter Name	DdsIPduHandleId		
Parent Container	DdsIPdu		
Description	Pdu Identifier assigned by Dds Module. Used by PduR for trasmission/reception APIs. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucIntegerParamDef (Symbolic Name generated for this parameter)		
Range	0 .. 65535		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

SWS Item	**[ECUC_Dds_00009]**		
Parameter Name	DdsIPduSide		
Parent Container	DdsIPdu		
Description	Identify the side of the IPdu. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucEnumerationParamDef		
Range	LOWER	Lower layer IPdu for Dds. **Tags:** atp.Status=draft	
	UPPER	Upper layer IPdu for Dds. **Tags:** atp.Status=draft	
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

SWS Item	**[ECUC_Dds_00010]**
Parameter Name	DdsIPduRef
Parent Container	DdsIPdu

▽

△

Description	Reference to the global Pdu. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	Reference to Pdu		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: local		

No Included Containers

10.2.3.2 DdsDomainParticipant

In the picture below, the UML diagram of DdsDomainParticipant container is shown

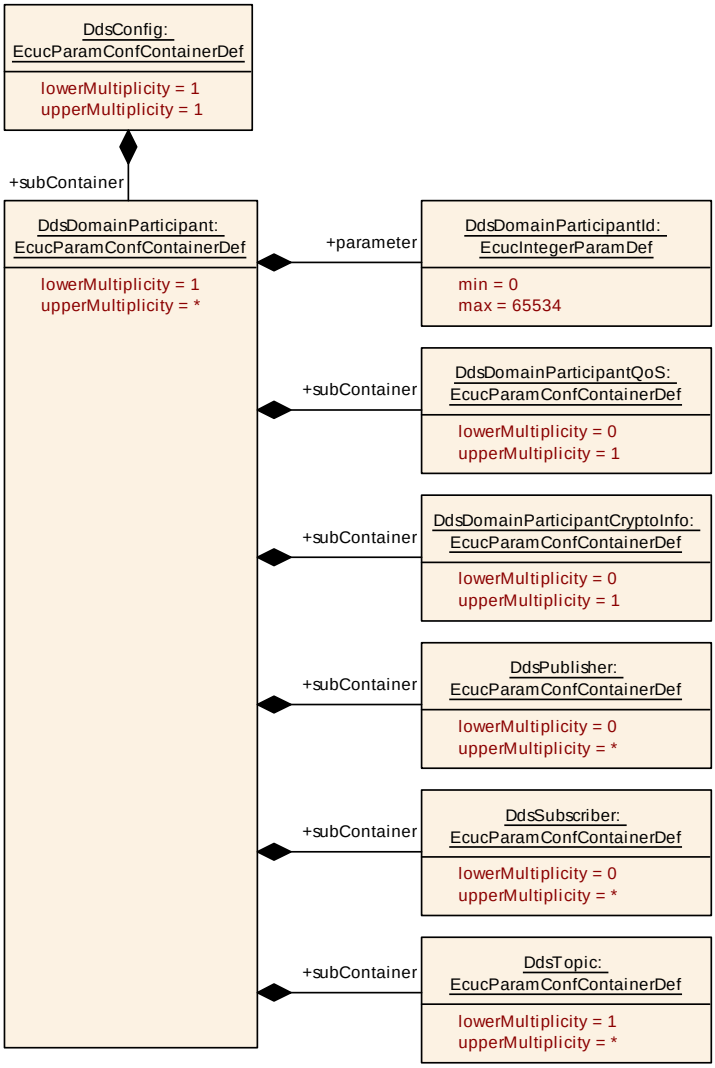

Figure 10.5: Dds Domain Participant

SWS Item	[ECUC_Dds_00012]		
Container Name	DdsDomainParticipant		
Parent Container	DdsConfig		
Description	This container represents the configuration of one single "Domain Participant" to which the current node belongs. One node can belong to more than one Domain Participant. One node can communicate only with other nodes beloning to the same Domain Participant. **Tags:** atp.Status=draft		
Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

SWS Item	[ECUC_Dds_00014]		
Parameter Name	DdsDomainParticipantId		
Parent Container	DdsDomainParticipant		
Description	Identifier of a Dds Domain Participant. Only entities that belong to the same domain participant can communicate each other. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucIntegerParamDef		
Range	0 .. 65534		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU dependency: inter-ECU - this value shall be shared between configurations.		

Included Containers		
Container Name	**Multiplicity**	**Scope / Dependency**
DdsDomainParticipantCryptoInfo	0..1	This container contains the configuration of the Crypto service to be used by Entities belonging to this DomainParticipant. If it is not present, it means that not security mechanism is supported. **Tags:** atp.Status=draft
DdsDomainParticipantQoS	0..1	This container represents the configuration of QoS supported by the Dds DomainParticipant. **Tags:** atp.Status=draft
DdsPublisher	0..*	This container represents the configuration of one Publisher. **Tags:** atp.Status=draft
DdsSubscriber	0..*	This container represents the configuration of a Subscriber. **Tags:** atp.Status=draft
DdsTopic	1..*	This container represents the configuration of one Topic. **Tags:** atp.Status=draft

10.2.3.2.1 DdsDomainParticipantQoS

SWS Item	[ECUC_Dds_00013]		
Container Name	DdsDomainParticipantQoS		
Parent Container	DdsDomainParticipant		
Description	This container represents the configuration of QoS supported by the Dds Domain Participant. **Tags:** atp.Status=draft		
Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

Included Containers		
Container Name	**Multiplicity**	**Scope / Dependency**
DdsEntityFactory	0..1	If present, this container indicates that Dds ENTITY_FACTORY QoS is supported by this DomainParticipant. **Tags:** atp.Status=draft
DdsUserData	0..1	If present, this container indicates that Dds USER_DATA QoS is supported by this DomainParticipant. **Tags:** atp.Status=draft

DdsUserData For description of this subcontainer, please refer to paragraph 10.2.3.3.1

DdsEntityFactory For description of this subcontainer, please refer to paragraph 10.2.3.3.20

10.2.3.2.2 DdsDomainParticipantCryptoInfo

In the picture below, the UML diagram of DdsDomainParticipantCryptoInfo container is shown

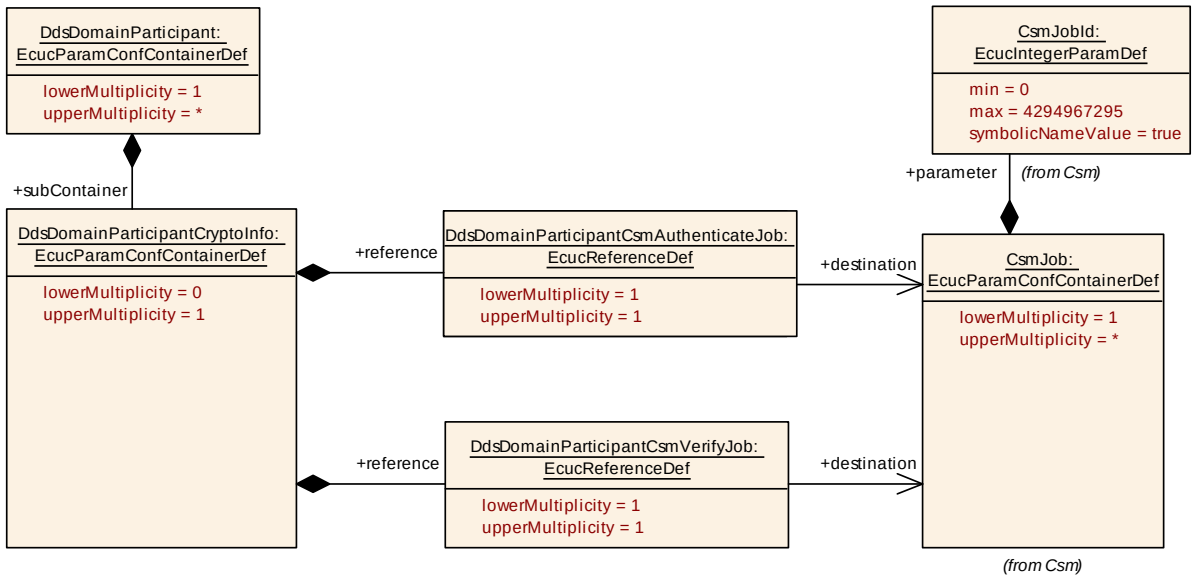

Figure 10.6: Dds DomainParticipant Crypto Info

SWS Item	**[ECUC_Dds_00015]**		
Container Name	DdsDomainParticipantCryptoInfo		
Parent Container	DdsDomainParticipant		
Description	This container contains the configuration of the Crypto service to be used by Entities belonging to this DomainParticipant. If it is not present, it means that not security mechanism is supported. **Tags:** atp.Status=draft		
Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

SWS Item	**[ECUC_Dds_00020]**		
Parameter Name	DdsDomainParticipantCsmAuthenticateJob		
Parent Container	DdsDomainParticipantCryptoInfo		
Description	The reference to the CSM job to be used to authenticate data. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	Reference to CsmJob		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

SWS Item	[ECUC_Dds_00021]		
Parameter Name	DdsDomainParticipantCsmVerifyJob		
Parent Container	DdsDomainParticipantCryptoInfo		
Description	The reference to the CSM job to be used to verify data. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	Reference to CsmJob		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

No Included Containers

10.2.3.2.3 DdsPublisher

In the picture below, the UML diagram of DdsPublisher container is shown

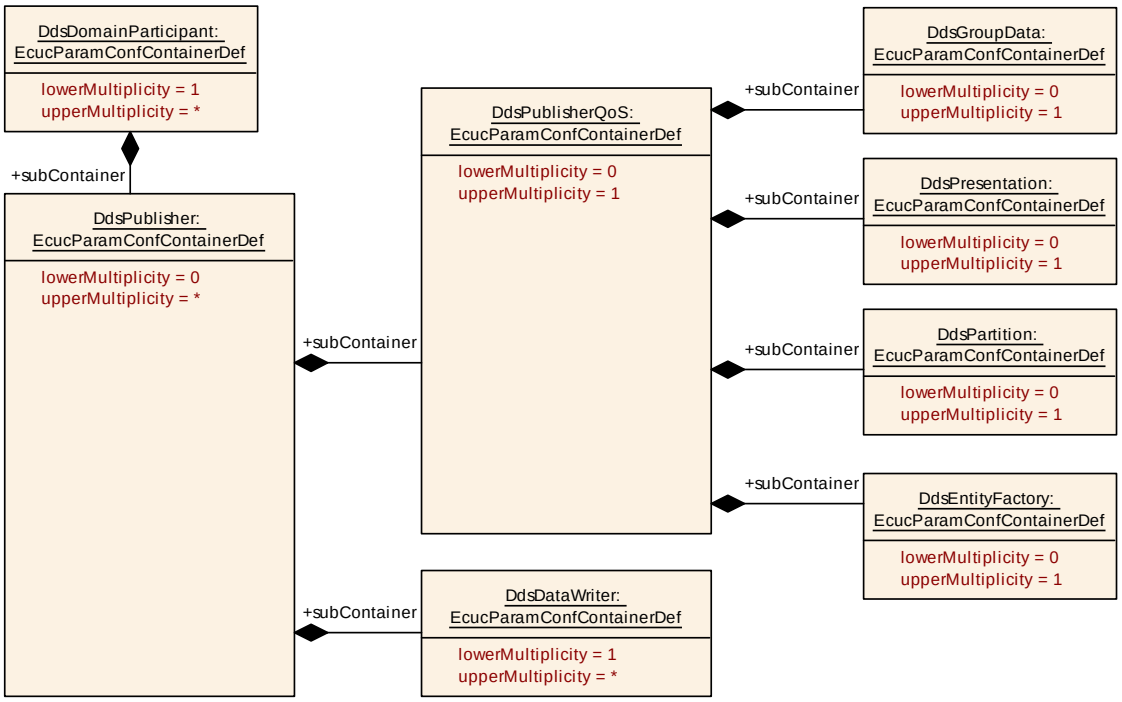

Figure 10.7: Dds Publisher

SWS Item	[ECUC_Dds_00016]
Container Name	DdsPublisher
Parent Container	DdsDomainParticipant

▽

△

Description	This container represents the configuration of one Publisher. **Tags:** atp.Status=draft		
Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

Included Containers		
Container Name	**Multiplicity**	**Scope / Dependency**
DdsDataWriter	1..*	This container represents the configuration of one data writer. One publisher can refer to one or more writer, but a writer can belong to one single publisher. **Tags:** atp.Status=draft
DdsPublisherQoS	0..1	This container represents the configuration of QoS Profiles related to the current Dds Publisher. **Tags:** atp.Status=draft

10.2.3.2.3.1 DdsPublisherQoS

SWS Item	**[ECUC_Dds_00022]**		
Container Name	DdsPublisherQoS		
Parent Container	DdsPublisher		
Description	This container represents the configuration of QoS Profiles related to the current Dds Publisher. **Tags:** atp.Status=draft		
Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

Included Containers		
Container Name	**Multiplicity**	**Scope / Dependency**
DdsEntityFactory	0..1	If present, this container indicates that Dds ENTITY_FACTORY QoS is supported. **Tags:** atp.Status=draft
DdsGroupData	0..1	If present, this container indicates that Dds GROUP_DATA QoS is supported. **Tags:** atp.Status=draft
DdsPartition	0..1	If present, this container indicates that Dds PARTITION QoS is supported. **Tags:** atp.Status=draft
DdsPresentation	0..1	If present, this container indicates that Dds PRESENTATION Qo S is supported. **Tags:** atp.Status=draft

DdsGroupData For description of this subcontainer, please refer to paragraph 10.2.3.3.3

DdsPresentation For description of this subcontainer, please refer to paragraph 10.2.3.3.6

DdsPartition For description of this subcontainer, please refer to paragraph 10.2.3.3.13

DdsEntityFactory For description of this subcontainer, please refer to paragraph 10.2.3.3.20

10.2.3.2.3.2 DdsDataWriter

In the picture below, the UML diagram of DdsDataWriter container is shown

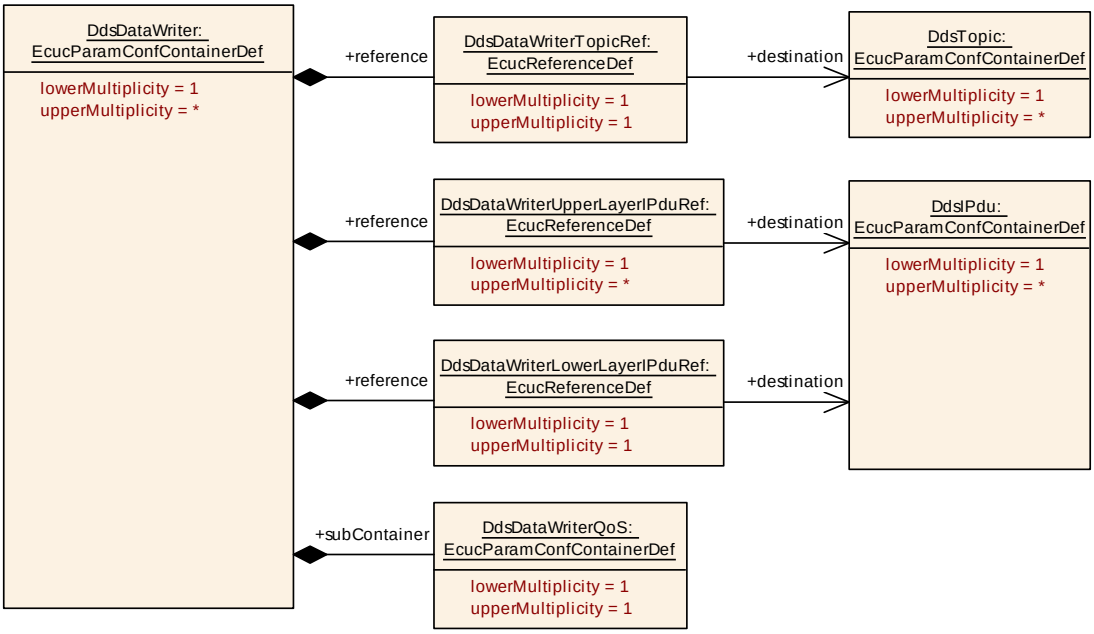

Figure 10.8: Dds DataWriter

SWS Item	**[ECUC_Dds_00023]**		
Container Name	DdsDataWriter		
Parent Container	DdsPublisher		
Description	This container represents the configuration of one data writer. One publisher can refer to one or more writer, but a writer can belong to one single publisher. **Tags:** atp.Status=draft		
Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

SWS Item	[ECUC_Dds_00031]		
Parameter Name	DdsDataWriterLowerLayerIPduRef		
Parent Container	DdsDataWriter		
Description	This reference selects the IPdu that the writer would use for lower layer. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	Reference to DdsIPdu		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

SWS Item	[ECUC_Dds_00029]		
Parameter Name	DdsDataWriterTopicRef		
Parent Container	DdsDataWriter		
Description	This reference selects the Topic on which the current Dds Writer wants to publish. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	Reference to DdsTopic		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

SWS Item	[ECUC_Dds_00030]		
Parameter Name	DdsDataWriterUpperLayerIPduRef		
Parent Container	DdsDataWriter		
Description	This reference selects the IPdu that the writer would use for upper layer. **Tags:** atp.Status=draft		
Multiplicity	1..*		
Type	Reference to DdsIPdu		
Post-Build Variant Multiplicity	false		
Post-Build Variant Value	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

Included Containers		
Container Name	**Multiplicity**	**Scope / Dependency**
DdsDataWriterQoS	1	This container represents the configuration of QoS Profiles related to the current DdsDataWriter. **Tags:** atp.Status=draft

DdsDataWriterQoS

SWS Item	[ECUC_Dds_00028]
Container Name	DdsDataWriterQoS
Parent Container	DdsDataWriter
Description	This container represents the configuration of QoS Profiles related to the current Dds DataWriter. **Tags:** atp.Status=draft
Configuration Parameters	

Included Containers		
Container Name	**Multiplicity**	**Scope / Dependency**
DdsDeadline	0..1	If present, this container indicates that Dds DEADLINE QoS is supported. **Tags:** atp.Status=draft
DdsDestinationOrder	0..1	If present, this container indicates that Dds DESTINATION_ ORDER QoS is supported. **Tags:** atp.Status=draft
DdsDurability	0..1	If present, this container indicates that Dds DURABILITY QoS is supported. **Tags:** atp.Status=draft
DdsDurabilityService	0..1	If present, this container indicates that Dds DURABILITY_ SERVICE QoS is supported. **Tags:** atp.Status=draft
DdsHistory	0..1	If present, this container indicates that Dds HISTORY QoS is supported. **Tags:** atp.Status=draft
DdsLatencyBudget	0..1	If present, this container indicates that Dds LATENCY_BUDGET QoS is supported. **Tags:** atp.Status=draft
DdsLifespan	0..1	If present, this container indicates that Dds LIFESPAN QoS is supported. **Tags:** atp.Status=draft
DdsLiveliness	0..1	If present, this container indicates that Dds LIVELINESS QoS is supported. **Tags:** atp.Status=draft
DdsOwnership	0..1	If present, this container indicates that Dds OWNERSHIP QoS is supported. **Tags:** atp.Status=draft
DdsOwnershipStrength	0..1	It describes the DDS OWNERSHIP_STRENGTH QoS policy. This policy specifies the value of the "strength" used to arbitrate among multiple DataWriter objects that attempt to modify the same instance of a data-object (identified by Topic + key). This policy only applies if the OWNERSHIP QoS policy is of kind EXCLUSIVE. The support of this policy is optional. It is mandatory if the OWNERSHIP QoS is supported and configured with kind = "Exclusive". This policy can be enabled only for DataWriters. **Tags:** atp.Status=draft

▽

△

Included Containers		
Container Name	**Multiplicity**	**Scope / Dependency**
DdsReliability	0..1	If present, this container indicates that Dds RELIABILITY QoS is supported. **Tags:** atp.Status=draft
DdsResourceLimits	0..1	If present, this container indicates that Dds RESOURCE_LIMITS QoS is supported. **Tags:** atp.Status=draft
DdsTransportPriority	0..1	If present, this container indicates that Dds TRANSPORT_ PRIORITY QoS is supported. **Tags:** atp.Status=draft
DdsUserData	0..1	If present, this container indicates that Dds USER_DATA QoS is supported. **Tags:** atp.Status=draft
DdsWriterDataLifecycle	0..1	It describes the DDS WRITER_DATA_LIFECYCLE QoS policy. It specifies the behavior of the DataWriter with regards to the lifecycle of the datainstances it manages. This policy can be enabled only for DataWriter. **Tags:** atp.Status=draft

DdsUserData For description of this subcontainer, please refer to paragraph 10.2.3.3.1

DdsDurability For description of this subcontainer, please refer to paragraph 10.2.3.3.4

DdsDurabilityService For description of this subcontainer, please refer to paragraph 10.2.3.3.5

DdsDeadline For description of this subcontainer, please refer to paragraph 10.2.3.3.7

DdsLatencyBudget For description of this subcontainer, please refer to paragraph 10.2.3.3.8

DdsOwnership For description of this subcontainer, please refer to paragraph 10.2.3.3.9

DdsOwnershipStrength For description of this subcontainer, please refer to paragraph 10.2.3.3.10

DdsLiveliness For description of this subcontainer, please refer to paragraph 10.2.3.3.11

DdsReliability For description of this subcontainer, please refer to paragraph 10.2.3.3.14

DdsTransportPriority For description of this subcontainer, please refer to paragraph 10.2.3.3.15

DdsLifespan For description of this subcontainer, please refer to paragraph 10.2.3.3.16

DdsDestinationOrder For description of this subcontainer, please refer to paragraph 10.2.3.3.17

DdsHistory For description of this subcontainer, please refer to paragraph 10.2.3.3.18

DdsResourceLimits For description of this subcontainer, please refer to paragraph 10.2.3.3.19

DdsWriterDataLifecycle For description of this subcontainer, please refer to paragraph 10.2.3.3.21

10.2.3.2.4 DdsSubscriber

In the picture below, the UML diagram of DdsSubscriber container is shown

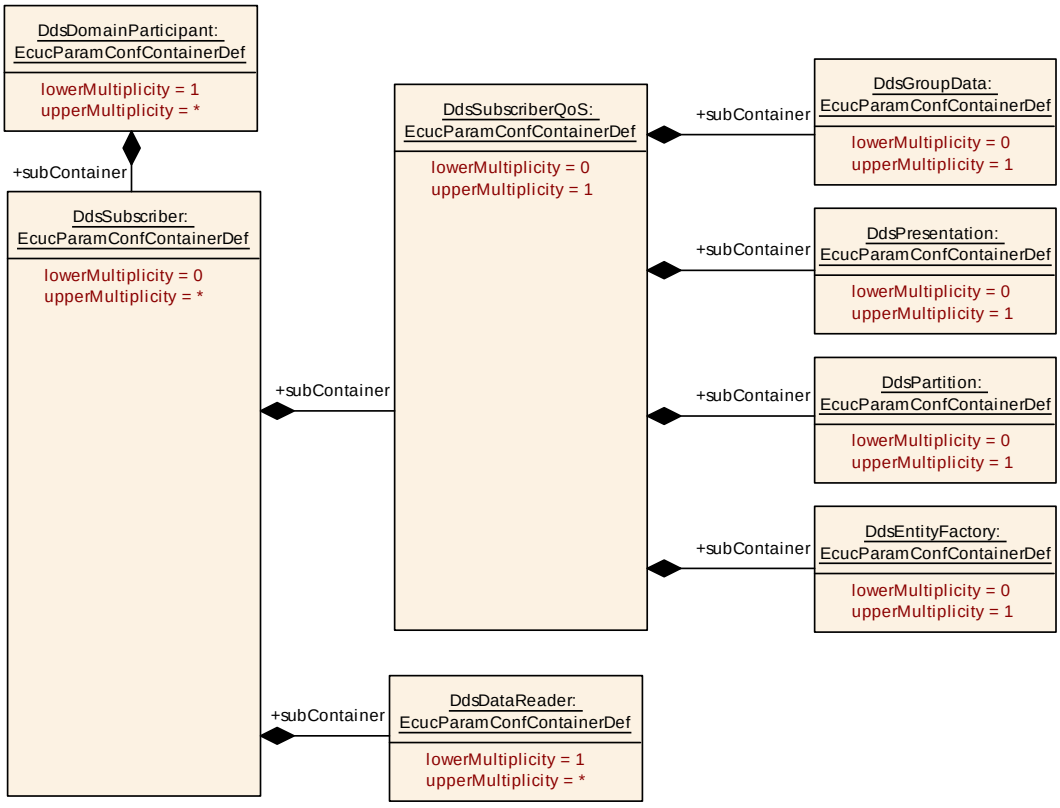

Figure 10.9: Dds Subscriber

SWS Item	[ECUC_Dds_00017]
Container Name	DdsSubscriber
Parent Container	DdsDomainParticipant
Description	This container represents the configuration of a Subscriber. **Tags:** atp.Status=draft
Post-Build Variant Multiplicity	false

△

Multiplicity Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

Included Containers		
Container Name	Multiplicity	Scope / Dependency
DdsDataReader	1..*	This container represents the configuration of one data reader. One subscriber can refer to one or more readers, but a reader can belong to one single subscriber. **Tags:** atp.Status=draft
DdsSubscriberQoS	0..1	This container represents the configuration of QoS Profiles related to the current Dds Subscriber. **Tags:** atp.Status=draft

10.2.3.2.4.1 DdsSubscriberQoS

SWS Item	[ECUC_Dds_00074]		
Container Name	DdsSubscriberQoS		
Parent Container	DdsSubscriber		
Description	This container represents the configuration of QoS Profiles related to the current Dds Subscriber. **Tags:** atp.Status=draft		
Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

Included Containers		
Container Name	Multiplicity	Scope / Dependency
DdsEntityFactory	0..1	If present, this container indicates that Dds ENTITY_FACTORY QoS is supported. **Tags:** atp.Status=draft
DdsGroupData	0..1	If present, this container indicates that Dds GROUP_DATA QoS is supported. **Tags:** atp.Status=draft
DdsPartition	0..1	If present, this container indicates that Dds PARTITION QoS is supported. **Tags:** atp.Status=draft
DdsPresentation	0..1	If present, this container indicates that Dds PRESENTATION Qo S is supported. **Tags:** atp.Status=draft

DdsGroupData For description of this subcontainer, please refer to paragraph 10.2.3.3.3

DdsPresentation For description of this subcontainer, please refer to paragraph 10.2.3.3.6

DdsPartition For description of this subcontainer, please refer to paragraph 10.2.3.3.13

DdsEntityFactory For description of this subcontainer, please refer to paragraph 10.2.3.3.20

10.2.3.2.4.2 DdsDataReader

In the picture below, the UML diagram of DdsDataReader container is shown

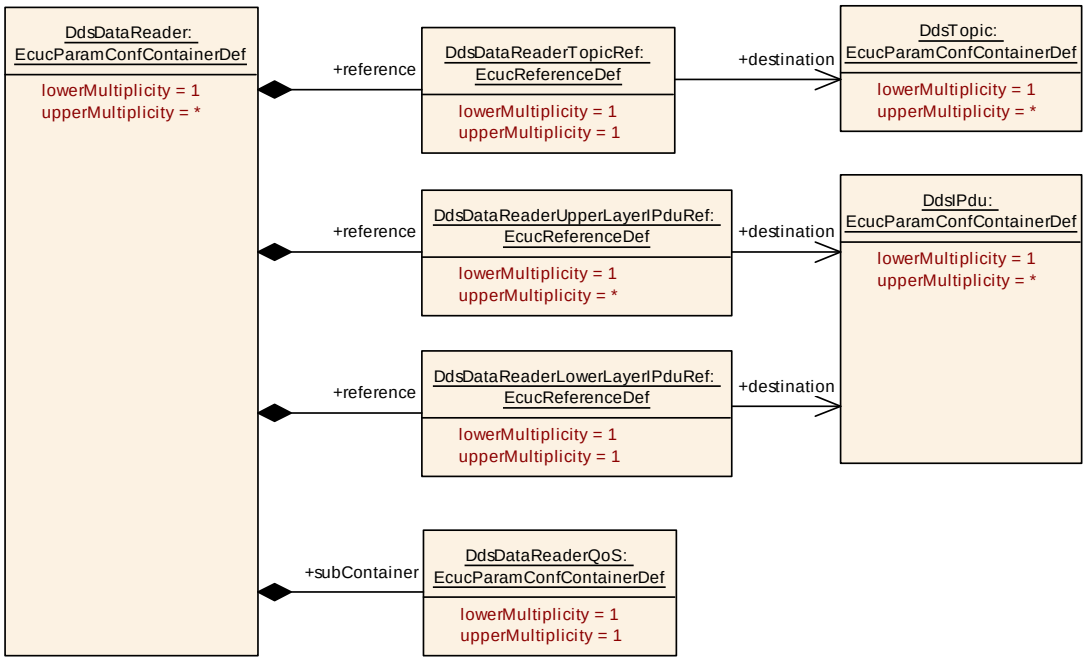

Figure 10.10: Dds DataReader

SWS Item	[ECUC_Dds_00075]		
Container Name	DdsDataReader		
Parent Container	DdsSubscriber		
Description	This container represents the configuration of one data reader. One subscriber can refer to one or more readers, but a reader can belong to one single subscriber. **Tags:** atp.Status=draft		
Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

SWS Item	[ECUC_Dds_00078]		
Parameter Name	DdsDataReaderLowerLayerIPduRef		
Parent Container	DdsDataReader		
Description	This reference selects the IPdu that the reader would use for lower layer. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	Reference to DdsIPdu		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

SWS Item	[ECUC_Dds_00076]		
Parameter Name	DdsDataReaderTopicRef		
Parent Container	DdsDataReader		
Description	This reference selects the Topic on which the current Dds Reader wants to publish. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	Reference to DdsTopic		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

SWS Item	[ECUC_Dds_00077]		
Parameter Name	DdsDataReaderUpperLayerIPduRef		
Parent Container	DdsDataReader		
Description	This reference selects the IPdu that the reader would use for upper layer. **Tags:** atp.Status=draft		
Multiplicity	1..*		
Type	Reference to DdsIPdu		
Post-Build Variant Multiplicity	false		
Post-Build Variant Value	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

Included Containers		
Container Name	**Multiplicity**	**Scope / Dependency**
DdsDataReaderQoS	1	This container represents the configuration of QoS Profiles related to the curret DdsDataReader. **Tags:** atp.Status=draft

DdsDataReaderQoS

SWS Item	[ECUC_Dds_00079]
Container Name	DdsDataReaderQoS
Parent Container	DdsDataReader
Description	This container represents the configuration of QoS Profiles related to the curret Dds DataReader. **Tags:** atp.Status=draft
Configuration Parameters	

Included Containers		
Container Name	**Multiplicity**	**Scope / Dependency**
DdsDeadline	0..1	If present, this container indicates that Dds DEADLINE QoS is supported. **Tags:** atp.Status=draft
DdsDestinationOrder	0..1	If present, this container indicates that Dds DESTINATION_ ORDER QoS is supported. **Tags:** atp.Status=draft
DdsHistory	0..1	If present, this container indicates that Dds HISTORY QoS is supported. **Tags:** atp.Status=draft
DdsLatencyBudget	0..1	If present, this container indicates that Dds LATENCY_BUDGET QoS is supported. **Tags:** atp.Status=draft
DdsLiveliness	0..1	If present, this container indicates that Dds LIVELINESS QoS is supported. **Tags:** atp.Status=draft
DdsOwnership	0..1	If present, this container indicates that Dds OWNERSHIP QoS is supported. **Tags:** atp.Status=draft
DdsReaderDataLifecycle	0..1	It describes the DDS READER_DATA_LIFECYCLE QoS policy. It specifies the behavior of the DataReader with regards to the lifecycle of the data-instances it manages. This policy can be enabled only for DataReader. **Tags:** atp.Status=draft
DdsReliability	0..1	If present, this container indicates that Dds RELIABILITY QoS is supported. **Tags:** atp.Status=draft
DdsResourceLimits	0..1	If present, this container indicates that Dds RESOURCE_LIMITS QoS is supported. **Tags:** atp.Status=draft
DdsTimeBasedFilter	0..1	It describes the DDS TIME_BASED_FILTER QoS policy. This policy allows a DataReader to indicate that it does not necessarily want to see all values of each instance published under the Topic. Rather, it wants to see at most one change every minimum_separation period. This policy can be enabled only for DataReaders. **Tags:** atp.Status=draft
DdsUserData	0..1	If present, this container indicates that Dds USER_DATA QoS is supported. **Tags:** atp.Status=draft

DdsUserData For description of this subcontainer, please refer to paragraph 10.2.3.3.1

DdsDurability For description of this subcontainer, please refer to paragraph 10.2.3.3.4

DdsDeadline For description of this subcontainer, please refer to paragraph 10.2.3.3.7

DdsLatencyBudget For description of this subcontainer, please refer to paragraph 10.2.3.3.8

DdsOwnership For description of this subcontainer, please refer to paragraph 10.2.3.3.9

DdsLiveliness For description of this subcontainer, please refer to paragraph 10.2.3.3.11

DdsTimeBasedFilter For description of this subcontainer, please refer to paragraph 10.2.3.3.12

DdsReliability For description of this subcontainer, please refer to paragraph 10.2.3.3.14

DdsDestinationOrder For description of this subcontainer, please refer to paragraph 10.2.3.3.17

DdsHistory For description of this subcontainer, please refer to paragraph 10.2.3.3.18

DdsResourceLimits For description of this subcontainer, please refer to paragraph 10.2.3.3.19

DdsReaderDataLifecycle For description of this subcontainer, please refer to paragraph 10.2.3.3.22

10.2.3.2.5 DdsTopic

In the picture below, the UML diagram of DdsTopic container is shown

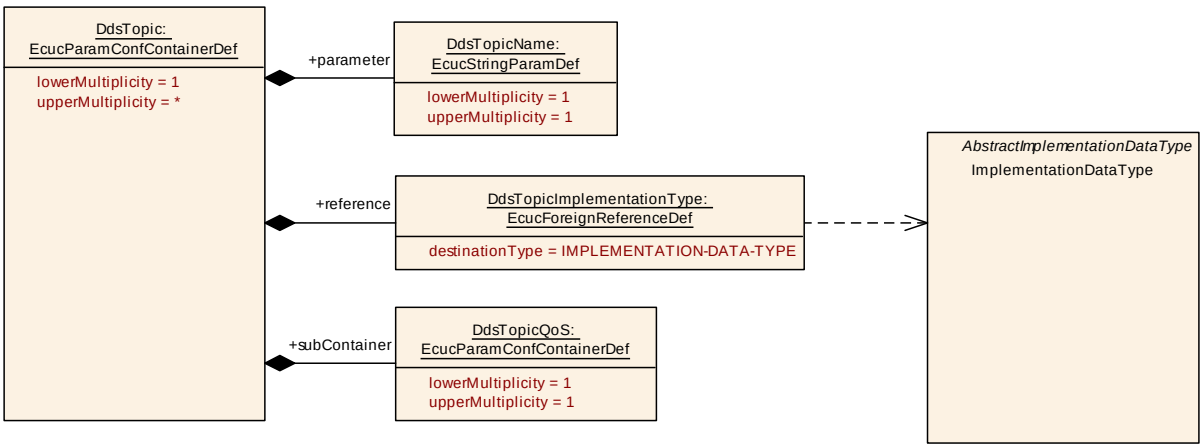

Figure 10.11: Dds Topic

SWS Item	[ECUC_Dds_00018]		
Container Name	DdsTopic		
Parent Container	DdsDomainParticipant		
Description	This container represents the configuration of one Topic. **Tags:** atp.Status=draft		
Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

SWS Item	[ECUC_Dds_00103]		
Parameter Name	DdsTopicName		
Parent Container	DdsTopic		
Description	Identify the name of the Topic. Publishers and subscribers intercommunication is based on topic name: only enteties that share the same topic can communicate each others. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucStringParamDef		
Default value	–		
Regular Expression	–		
Post-Build Variant Multiplicity	false		
Post-Build Variant Value	false		
Multiplicity Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU dependency: inter-ECU - this value shall be shared between configurations.		

SWS Item	[ECUC_Dds_00104]		
Parameter Name	DdsTopicImplementationType		
Parent Container	DdsTopic		
Description	This reference selects the ImplementationDataType the topic is related. A Topic is used to publish a well-defined data type, described by the referenced ImplementationData Type. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	Foreign reference to IMPLEMENTATION-DATA-TYPE		
Post-Build Variant Value	false		
Value Configuration Class	Pre-compile time	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

Included Containers		
Container Name	**Multiplicity**	**Scope / Dependency**
DdsTopicQoS	1	This container contains the configuration of the QoS supported by the DdsTopic **Tags:** atp.Status=draft

10.2.3.2.5.1 DdsTopicQoS

SWS Item	[ECUC_Dds_00102]
Container Name	DdsTopicQoS
Parent Container	DdsTopic
Description	This container contains the configuration of the QoS supported by the DdsTopic **Tags:** atp.Status=draft
Configuration Parameters	

Included Containers		
Container Name	**Multiplicity**	**Scope / Dependency**
DdsDeadline	0..1	If present, this container indicates that Dds DEADLINE QoS is supported. **Tags:** atp.Status=draft
DdsDestinationOrder	0..1	If present, this container indicates that Dds DESTINATION_ ORDER QoS is supported. **Tags:** atp.Status=draft
DdsDurability	0..1	If present, this container indicates that Dds DURABILITY QoS is supported. **Tags:** atp.Status=draft
DdsDurabilityService	0..1	If present, this container indicates that Dds DURABILITY_ SERVICE QoS is supported. **Tags:** atp.Status=draft
DdsHistory	0..1	If present, this container indicates that Dds HISTORY QoS is supported. **Tags:** atp.Status=draft
DdsLatencyBudget	0..1	If present, this container indicates that Dds LATENCY_BUDGET QoS is supported. **Tags:** atp.Status=draft
DdsLifespan	0..1	If present, this container indicates that Dds LIFESPAN QoS is supported. **Tags:** atp.Status=draft
DdsLiveliness	0..1	If present, this container indicates that Dds LIVELINESS QoS is supported. **Tags:** atp.Status=draft
DdsOwnership	0..1	If present, this container indicates that Dds OWNERSHIP QoS is supported. **Tags:** atp.Status=draft
DdsReliability	0..1	If present, this container indicates that Dds RELIABILITY QoS is supported. **Tags:** atp.Status=draft

▽

△

Included Containers		
Container Name	**Multiplicity**	**Scope / Dependency**
DdsResourceLimits	0..1	If present, this container indicates that Dds RESOURCE_LIMITS QoS is supported. **Tags:** atp.Status=draft
DdsTopicData	0..1	It describes the DDS TOPIC_DATA QoS policy. The purpose of this QoS is to allow the application to attach additional information to the created Topic. This policy can be enabled only for Topic entities. **Tags:** atp.Status=draft
DdsTransportPriority	0..1	If present, this container indicates that Dds TRANSPORT_PRIORITY QoS is supported. **Tags:** atp.Status=draft

DdsTopicData For description of this subcontainer, please refer to paragraph 10.2.3.3.2

DdsDurability For description of this subcontainer, please refer to paragraph 10.2.3.3.4

DdsDurabilityService For description of this subcontainer, please refer to paragraph 10.2.3.3.5

DdsDeadline For description of this subcontainer, please refer to paragraph 10.2.3.3.7

DdsLatencyBudget For description of this subcontainer, please refer to paragraph 10.2.3.3.8

DdsOwnership For description of this subcontainer, please refer to paragraph 10.2.3.3.9

DdsLiveliness For description of this subcontainer, please refer to paragraph 10.2.3.3.11

DdsReliability For description of this subcontainer, please refer to paragraph 10.2.3.3.14

DdsTransportPriority For description of this subcontainer, please refer to paragraph 10.2.3.3.15

DdsLifespan For description of this subcontainer, please refer to paragraph 10.2.3.3.16

DdsDestinationOrder For description of this subcontainer, please refer to paragraph 10.2.3.3.17

DdsHistory For description of this subcontainer, please refer to paragraph 10.2.3.3.18

DdsResourceLimits For description of this subcontainer, please refer to paragraph 10.2.3.3.19

10.2.3.3 DdsQoSPolicies

Note: This chapter is intendend to describe all the possible QoS policies. Each of them can be applied only to specific entity types.

The container DdsQoSPolicies does not exist actually, there is a specific Dds<Entity_type>QoS subcontainer for each entity type that supports QoS policies. For each entity type, in the related chapter, the correct Dds<Entity_type>QoS is described.

10.2.3.3.1 DdsUserData

[CP_SWS_Dds_01001]{DRAFT} **DDS USER_DATA semantics** ⌈If DdsUserData (the container used for DDS USER_DATA QoS policy) is configured for a specific entity (DdsDomainParticipants, DdsDataReaders or DdsDataWriters), the Dds BSW shall provide user defined information for this entity to other entities in the same DdsDomainParticipant. User data information would be distributed in the context of a DomainParticipant only by means of build-in topics, not with every exchanged message.

The DdsUserDataValue buffer shall be statically configured according an external agreement between parties and it will be never modified at runtime. Being statically configured and not accessible from the application, which can not modify its content, no API is required.

Note: the configuration and meaning of USER_DATA QoS is vendor specific, it can be used to exchange any kind of information between entities. For any details on this QoS topic, the "USER_DATA" paragraph of [1] shall be taken as reference.

⌋*(FO_RS_Dds_00005)*

SWS Item			
Container Name	DdsUserData		
Parent Container	DdsDataReaderQoS, DdsDataWriterQoS, DdsDomainParticipantQoS		
Description	It describes the DDS USER_DATA QoS policy. The purpose of this QoS is to allow the application to attach additional information to the related object. One possible use of this QoS is (but not limited to) to attach security credentials or some other information that can be used by the remote application to authenticate the source. This policy can be enabled only for DomainParticipants, Data Readers or DataWriters. **Tags:** atp.Status=draft		
Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

SWS Item	[ECUC_Dds_00124]		
Parameter Name	DdsUserDataValue		
Parent Container	DdsUserData		
Description	Defines the string used as USER_DATA value. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucStringParamDef		
Default value	–		
Regular Expression	–		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

No Included Containers

10.2.3.3.2 DdsTopicData

SWS Item	[ECUC_Dds_00106]		
Container Name	DdsTopicData		
Parent Container	DdsTopicQoS		
Description	It describes the DDS TOPIC_DATA QoS policy. The purpose of this QoS is to allow the application to attach additional information to the created Topic. This policy can be enabled only for Topic entities. **Tags:** atp.Status=draft		
Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

SWS Item	[ECUC_Dds_00126]		
Parameter Name	DdsTopicDataValue		
Parent Container	DdsTopicData		
Description	Defines the string used as TOPIC_DATA value. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucStringParamDef		
Default value	–		
Regular Expression	–		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants

△

	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

No Included Containers

10.2.3.3.3 DdsGroupData

SWS Item			
Container Name	DdsGroupData		
Parent Container	DdsPublisherQoS, DdsSubscriberQoS		
Description	If present, this container indicates that Dds GROUP_DATA QoS is supported. **Tags:** atp.Status=draft		
Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

SWS Item	**[ECUC_Dds_00125]**		
Parameter Name	DdsGroupDataValue		
Parent Container	DdsGroupData		
Description	Defines the string used as GROUP_DATA value. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucStringParamDef		
Default value	–		
Regular Expression	–		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

No Included Containers

10.2.3.3.4 DdsDurability

SWS Item	
Container Name	DdsDurability
Parent Container	DdsDataWriterQoS, DdsTopicQoS

▽

△

Description	It describes the DDS DURABILITY QoS policy. This policy can be enabled only for Topic, DataReaders or DataWriters entities This QoS policy controls whether the Service will actually make data available to late-joining readers. **Tags:** atp.Status=draft		
Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

SWS Item	**[ECUC_Dds_00035]**	
Parameter Name	DdsDurabilityKind	
Parent Container	DdsDurability	
Description	This reference selects the Topic on which the current Dds Writer wants to publish. Note: The value offered is considered compatible with the value requested if and only if the inequality "offered kind >= requested kind" evaluates to 'TRUE.' For the purposes of this inequality, the values of DURABILITY kind are considered ordered such that VOLATILE < TRANSIENT_LOCAL < TRANSIENT < PERSISTENT. If kind = TRANSIENT_LOCAL, TRANSIENT or PERSISTENT, the presence of DURABILITY_SERVICE QoS policy is mandatory. **Tags:** atp.Status=draft	
Multiplicity	1	
Type	EcucEnumerationParamDef	
Range	PERSISTENT	Data is kept on permanent storage, so that they can outlive a system session. Support for PERSISTENT is optional. **Tags:** atp.Status=draft
	TRANSIENT	The service is only required to keep the data in memory and not in permanent storage; but the data is not tied to the lifecycle of the DataWriter and will, in general, survive it. Support for TRANSIENT kind is optional. **Tags:** atp.Status=draft
	TRANSIENT_LOCAL	The service is only required to keep the data in the memory of the DataWriter that wrote the data and the data is not required to survive the Data Writer. **Tags:** atp.Status=draft
	VOLATILE	The Service does not need to keep any samples of data-instances on behalf of any DataReader that is not known by the DataWriter at the time the instance is written. In other words the Service will only attempt to provide the data to existing subscribers. **Tags:** atp.Status=draft
Default value	VOLATILE	
Post-Build Variant Value	false	
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	

▽

△

	Post-build time	–	
Scope / Dependency	scope: ECU		

No Included Containers

10.2.3.3.5 DdsDurabilityService

SWS Item			
Container Name	DdsDurabilityService		
Parent Container	DdsDataWriterQoS, DdsTopicQoS		
Description	It describes the DDS DURABILITY_SERVICE QoS policy. This policy can be enabled only for Topic or DataWriters entities This policy specifies the configuration of the durability service. That is, the service that implements the DURABILITY kind of TRANSIENT and PERSISTENT. **Tags:** atp.Status=draft		
Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

SWS Item	**[ECUC_Dds_00037]**		
Parameter Name	DdsDurabilityServiceCleanupDelay		
Parent Container	DdsDurabilityService		
Description	This parameter controls when the service is able to remove all information regarding a data-instance. Time given in seconds. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucFloatParamDef		
Range	[0 .. 65.534]		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

SWS Item	**[ECUC_Dds_00119]**
Parameter Name	DdsDurabilityServiceHistoryDepth
Parent Container	DdsDurabilityService
Description	This parameter controls the number of sample to store within the durability service. **Tags:** atp.Status=draft
Multiplicity	1

▽

△

Type	EcucIntegerParamDef		
Range	0 .. 65534		
Default value	1		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

SWS Item	**[ECUC_Dds_00038]**	
Parameter Name	DdsDurabilityServiceHistoryKind	
Parent Container	DdsDurabilityService	
Description	This parameter controls the HISTORY QoS of the fictitious DataReader that stores the data within the durability service. **Tags:** atp.Status=draft	
Multiplicity	1	
Type	EcucEnumerationParamDef	
Range	KEEP_ALL	The service is only required to keep the data in the memory of the DataWriter that wrote the data and the data is not required to survive the Data Writer. **Tags:** atp.Status=draft
	KEEP_LAST	The Service does not need to keep any samples of data-instances on behalf of any DataReader that is not known by the DataWriter at the time the instance is written. In other words the Service will only attempt to provide the data to existing subscribers. **Tags:** atp.Status=draft
Default value	KEEP_LAST	
Post-Build Variant Value	false	
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU	

SWS Item	**[ECUC_Dds_00121]**		
Parameter Name	DdsDurabilityServiceMaxInstances		
Parent Container	DdsDurabilityService		
Description	Please refer to chapt 2.2.3.5. of DDS Spec. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucIntegerParamDef		
Range	0 .. 65534		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	

▽

△

Scope / Dependency	scope: ECU

SWS Item	**[ECUC_Dds_00120]**		
Parameter Name	DdsDurabilityServiceMaxSamples		
Parent Container	DdsDurabilityService		
Description	This parameter controls the maximum number of samples to store within the durability service. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucIntegerParamDef		
Range	0 .. 65534		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

SWS Item	**[ECUC_Dds_00122]**		
Parameter Name	DdsDurabilityServiceMaxSamplesPerInstance		
Parent Container	DdsDurabilityService		
Description	Please refer to chapt 2.2.3.5. of DDS Spec. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucIntegerParamDef		
Range	0 .. 65534		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

No Included Containers

10.2.3.3.6 DdsPresentation

SWS Item	
Container Name	DdsPresentation
Parent Container	DdsPublisherQoS, DdsSubscriberQoS

▽

△

Description	It describes the PRESENTATION QoS policy. This policy specifies how the samples representing changes to data instances are presented to the subscribing application. This policy can be enabled only for Publishers or Subscribers entities. For any details, please refer to chapter 2.2.3.6 of DDS specification **Tags:** atp.Status=draft		
Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

SWS Item	**[ECUC_Dds_00069]**		
Parameter Name	DdsPresentationAccessScope		
Parent Container	DdsPresentation		
Description	This parameter determines the largest scope spanning the entities for which the order and coherency of changes can be preserved. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucEnumerationParamDef		
Range	GROUP	See 2.2.3.6 of DDS Spec for details. **Tags:** atp.Status=draft	
	INSTANCE	See 2.2.3.6 of DDS Spec for details. **Tags:** atp.Status=draft	
	TOPIC	See 2.2.3.6 of DDS Spec for details. **Tags:** atp.Status=draft	
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

SWS Item	**[ECUC_Dds_00070]**		
Parameter Name	DdsPresentationCoherentAccess		
Parent Container	DdsPresentation		
Description	This parameter controls whether coherent access is supported within the scope access_scope. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucBooleanParamDef		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: local		

SWS Item	[ECUC_Dds_00071]		
Parameter Name	DdsPresentationOrderedAccess		
Parent Container	DdsPresentation		
Description	This parameter controls whether ordered access is supported within the scope access_scope. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucBooleanParamDef		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: local		

No Included Containers

10.2.3.3.7 DdsDeadline

SWS Item			
Container Name	DdsDeadline		
Parent Container	DdsDataReaderQoS, DdsDataWriterQoS, DdsTopicQoS		
Description	It describes the DDS DEADLINE QoS policy. This policy is useful for cases where a Topic is expected to have each instance updated periodically. This policy can be enabled only for Topics, DataReaders and DataWriters. **Tags:** atp.Status=draft		
Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

SWS Item	[ECUC_Dds_00040]
Parameter Name	DdsDeadlinePeriod
Parent Container	DdsDeadline
Description	DataReader expects a new sample updating the value of each instance at least once every deadline period. DataWriter indicates that the application commits to write a new value for each instance managed by the DataWriter at least once every deadline period. It is inconsistent for a DataReader to have a DEADLINE period less than its TIME_BASED_FILTER's minimum_separation. Time given in seconds. **Tags:** atp.Status=draft
Multiplicity	1
Type	EcucFloatParamDef

▽

△

Range	[0 .. 65.534]		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

No Included Containers

10.2.3.3.8 DdsLatencyBudget

SWS Item			
Container Name	DdsLatencyBudget		
Parent Container	DdsDataReaderQoS, DdsDataWriterQoS, DdsTopicQoS		
Description	It describes the DDS LATENCY_BUDGET QoS policy. This policy provides a means for the application to indicate to the middleware the "urgency" of the data-communication. By having a non-zero duration the Service can optimize its internal operation. This policy is considered a hint. There is no specified mechanism as to how the service should take advantage of this hint. This policy can be enabled only for Topics, DataReaders and DataWriters. **Tags:** atp.Status=draft		
Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

SWS Item	**[ECUC_Dds_00042]**		
Parameter Name	DdsLatencyBudgetDuration		
Parent Container	DdsLatencyBudget		
Description	The default value of the duration is zero indicating that the delay should be minimized. Time given in seconds. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucFloatParamDef		
Range	[0 .. 65.534]		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

No Included Containers

10.2.3.3.9 DdsOwnership

SWS Item			
Container Name	DdsOwnership		
Parent Container	DdsDataReaderQoS, DdsDataWriterQoS, DdsTopicQoS		
Description	It describes the DDS OWNERSHIP QoS policy. This policy specifies whether it is allowed for multiple DataWriters to write the same instance of the data and if so, how these modifications should be arbitrated. The support of this policy is optional. This policy can be enabled only for Topics, DataReaders and DataWriters. **Tags:** atp.Status=draft		
Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

SWS Item	**[ECUC_Dds_00044]**		
Parameter Name	DdsOwnershipKind		
Parent Container	DdsOwnership		
Description	This parameter determines the kind of ownership. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucEnumerationParamDef		
Range	EXCLUSIVE	Indicates each instance can only be owned by one DataWriter, but the owner of an nstance can change dynamically. The selection of the owner is controlled by the setting of the OWNERSHIP_STRENGTH QoS policy. The owner is always set to be the highest strength DataWriter object among the ones currently "active. **Tags:** atp.Status=draft	
	SHARED	Indicates shared ownership for each instance. Multiple writers are allowed to update the same instance and all the updates are made available to the readers. In other words there is no concept of an "owner" for the instances. This is the default behavior if the OWNERSHIP QoS policy is not specified or supported. **Tags:** atp.Status=draft	
Default value	SHARED		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

No Included Containers

10.2.3.3.10 DdsOwnershipStrength

SWS Item	[ECUC_Dds_00045]		
Container Name	DdsOwnershipStrength		
Parent Container	DdsDataWriterQoS		
Description	It describes the DDS OWNERSHIP_STRENGTH QoS policy. This policy specifies the value of the "strength" used to arbitrate among multiple Data Writer objects that attempt to modify the same instance of a data-object (identified by Topic + key). This policy only applies if the OWNERSHIP QoS policy is of kind EXCLUSIVE. The support of this policy is optional. It is mandatory if the OWNERSHIP QoS is supported and configured with kind = "Exclusive". This policy can be enabled only for DataWriters. **Tags:** atp.Status=draft		
Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

SWS Item	[ECUC_Dds_00046]	
Parameter Name	DdsOwnershipStrengthValue	
Parent Container	DdsOwnershipStrength	
Description	This parameters specifies the value of "strength" used. **Tags:** atp.Status=draft	
Multiplicity	1	
Type	EcucIntegerParamDef	
Range	0 .. 65534	
Default value	0	
Post-Build Variant Value	false	
Value Configuration Class	**Pre-compile time**	X All Variants
	Link time	–
	Post-build time	–
Scope / Dependency	scope: ECU	

No Included Containers

10.2.3.3.11 DdsLiveliness

SWS Item	
Container Name	DdsLiveliness
Parent Container	DdsDataReaderQoS, DdsDataWriterQoS, DdsTopicQoS

△

Description	It describes the DDS LIVELINESS QoS policy. This policy determines the mechanism and parameters used by the application to determine whether an Entity is "active" (alive). This policy can be enabled only for Topics, DataReaders and DataWriters. **Tags:** atp.Status=draft		
Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

SWS Item	**[ECUC_Dds_00049]**		
Parameter Name	DdsLivelinessLeaseDuration		
Parent Container	DdsLiveliness		
Description	The DataReader requests that liveliness of the writers is maintained by the requested means and loss of liveliness is detected with delay not to exceed the lease_duration. The DataWriter commits to signalling its liveliness using the stated means at intervals not to exceed the lease_duration. Time given in seconds. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucFloatParamDef		
Range	[0 .. 65.534]		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

SWS Item	**[ECUC_Dds_00048]**	
Parameter Name	DdsLivenessKind	
Parent Container	DdsLiveliness	
Description	This parameter determines the kind of liveness. **Tags:** atp.Status=draft	
Multiplicity	1	
Type	EcucEnumerationParamDef	
Range	AUTOMATIC	The infrastructure will automatically signal liveliness for the DataWriters at least as often as required by the lease_duration. **Tags:** atp.Status=draft
	MANUAL_BY_PARTICIPANT	The Service will assume that as long as at least one Entity within the DomainParticipant has asserted its liveliness the other Entities in that same DomainParticipant are also alive. **Tags:** atp.Status=draft

▽

△

	MANUAL_BY_TOPIC	The Service will only assume liveliness of the DataWriter if the application has asserted liveliness of that DataWriter itself. **Tags:** atp.Status=draft	
Default value	AUTOMATIC		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

No Included Containers

10.2.3.3.12 DdsTimeBasedFilter

SWS Item	**[ECUC_Dds_00087]**		
Container Name	DdsTimeBasedFilter		
Parent Container	DdsDataReaderQoS		
Description	It describes the DDS TIME_BASED_FILTER QoS policy. This policy allows a DataReader to indicate that it does not necessarily want to see all values of each instance published under the Topic. Rather, it wants to see at most one change every minimum_separation period. This policy can be enabled only for DataReaders. **Tags:** atp.Status=draft		
Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

SWS Item	**[ECUC_Dds_00088]**		
Parameter Name	DdsTimeBasedFilterMinimumSeparation		
Parent Container	DdsTimeBasedFilter		
Description	The filter states that the DataReader does not want to receive more than one value each minimum_separation. By default minimum_separation=0.0 indicating DataReader is potentially interested in all values. Time given in seconds. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucFloatParamDef		
Range	[0 .. 65.534]		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	

▽

△

	Post-build time	–	
Scope / Dependency	scope: ECU		

No Included Containers

10.2.3.3.13 DdsPartition

SWS Item			
Container Name	DdsPartition		
Parent Container	DdsPublisherQoS, DdsSubscriberQoS		
Description	It describes the DDS PARTITION QoS policy. This policy allows the introduction of a logical partition concept inside the partition induced by a domain. This policy can be enabled only for Publishers or Subscribers. **Tags:** atp.Status=draft		
Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

SWS Item	**[ECUC_Dds_00072]**		
Parameter Name	DdsPartitionName		
Parent Container	DdsPartition		
Description	Set of strings that introduces a logical partition among the topics visible by the Publisher and Subscriber. A DataWriter within a Publisher only communicates with a DataReader in a Subscriber if (in addition to matching the Topic and having compatible QoS) the Publisher and Subscriber have a common partition name string. The empty string ("") is considered a valid partition that is matched with other partition names using the same rules of string matching and regular-expression matching used for any other partition name (see 2.2.3.13) The default value for the PARTITION QoS is a zero-length sequence. The zero-length sequence is treated as a special value equivalent to a equence containing a single element consisting of the empty string. **Tags:** atp.Status=draft		
Multiplicity	1..65534		
Type	EcucStringParamDef		
Default value	–		
Regular Expression	–		
Post-Build Variant Multiplicity	false		
Post-Build Variant Value	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	

▽

△

	Post-build time	–	
Scope / Dependency	scope: ECU		

No Included Containers

10.2.3.3.14 DdsReliability

SWS Item			
Container Name	DdsReliability		
Parent Container	DdsDataReaderQoS, DdsDataWriterQoS, DdsTopicQoS		
Description	It Indicates the level of reliability offered/ requested by the Service. This policy can be enabled only for Topics, DataReaders and DataWriters. **Tags:** atp.Status=draft		
Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

SWS Item	**[ECUC_Dds_00051]**		
Parameter Name	DdsReliabilityKind		
Parent Container	DdsReliability		
Description	This parameter determines the kind of reliability. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucEnumerationParamDef		
Range	BEST_EFFORT	Indicates that it is acceptable to not retry propagation of any samples. **Tags:** atp.Status=draft	
	RELIABLE	Specifies the Service will attempt to deliver all samples in its history. Missed samples may be retried. **Tags:** atp.Status=draft	
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

SWS Item	**[ECUC_Dds_00052]**
Parameter Name	DdsReliabilityMaxBlockingTime
Parent Container	DdsReliability

▽

△

Description	The value of the max_blocking_time indicates the maximum time the operation Data Writer::write is allowed to block if the DataWriter does not have space to store the value written. Time given in seconds. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucFloatParamDef		
Range	[0 .. 65.534]		
Default value	0.1		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

No Included Containers

10.2.3.3.15 DdsTransportPriority

SWS Item			
Container Name	DdsTransportPriority		
Parent Container	DdsDataWriterQoS, DdsTopicQoS		
Description	It describes the DDS TRANSPORT_PRIORITY QoS policy. The purpose of this QoS is to allow the application to take advantage of transports capable of sending messages with different priorities. The policy depends on the ability of the underlying transports to set a priority on the messages they send. This policy can be enabled only for Topics and DataWriters. **Tags:** atp.Status=draft		
Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

SWS Item	**[ECUC_Dds_00054]**		
Parameter Name	DdsTransportPriorityValue		
Parent Container	DdsTransportPriority		
Description	This parameter represents the priority to be set. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucIntegerParamDef		
Range	0 .. 65534		
Default value	0		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants

▽

△

	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

No Included Containers

10.2.3.3.16 DdsLifespan

SWS Item			
Container Name	DdsLifespan		
Parent Container	DdsDataWriterQoS, DdsTopicQoS		
Description	It describes the DDS LIFESPAN QoS policy. The purpose of this QoS is to allow the application to take advantage of transports capable of sending messages with different priorities. The policy depends on the ability of the underlying transports to set a priority on the messages they send. This policy can be enabled only for Topics and DataWriters. **Tags:** atp.Status=draft		
Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

SWS Item	**[ECUC_Dds_00056]**		
Parameter Name	DdsLifespanDuration		
Parent Container	DdsLifespan		
Description	Specifies the maximum duration of validity of the data written by the DataWriter. Time given in seconds. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucFloatParamDef		
Range	[0 .. 65.534]		
Default value	–		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

No Included Containers

10.2.3.3.17 DdsDestinationOrder

SWS Item			
Container Name	DdsDestinationOrder		
Parent Container	DdsDataReaderQoS, DdsDataWriterQoS, DdsTopicQoS		
Description	It describes the DDS DESTINATION_ORDER QoS policy. This policy controls how each subscriber resolves the final value of a data instance that is written by multiple DataWriter objects (which may be associated with different Publisher objects) running on different nodes. This policy can be enabled only for DataWriters and DataReaders. **Tags:** atp.Status=draft		
Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

SWS Item	**[ECUC_Dds_00058]**		
Parameter Name	DdsDestinationOrderKind		
Parent Container	DdsDestinationOrder		
Description	This parameter controls the criteria used to determine the logical order among changes made by Publisher entities to the same instance of data (i.e., matching Topic and key). **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucEnumerationParamDef		
Range	BY_RECEPTION_TIMESTAMP	It indicates that data is ordered based on the reception time at each Subscriber. Since each subscriber may receive the data at different times there is no guaranteed that the changes will be seen in the same order. Consequently, it is possible for each subscriber to end up with a different final value for the data. **Tags:** atp.Status=draft	
	BY_SOURCE_TIMESTAMP	Indicates that data is ordered based on a timestamp placed at the source (by the Service or by the application). In any case this guarantees a consistent final value for the data in all subscribers. **Tags:** atp.Status=draft	
Default value	BY_RECEPTION_TIMESTAMP		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

No Included Containers

10.2.3.3.18 DdsHistory

SWS Item			
Container Name	DdsHistory		
Parent Container	DdsDataReaderQoS, DdsDataWriterQoS, DdsTopicQoS		
Description	It describes the DDS HISTORY QoS policy. Specifies the behavior of the Service in the case where the value of a sample changes (one or more times) before it can be successfully communicated to one or more existing subscribers. This QoS policy controls whether the Service should deliver only the most recent value, attempt to deliver all intermediate values, or do something in between. This policy can be enabled only for DataWriters, DataReaders and Topics. **Tags:** atp.Status=draft		
Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

SWS Item	**[ECUC_Dds_00060]**		
Parameter Name	DdsHistoryKind		
Parent Container	DdsHistory		
Description	This parameter controls the criteria used to determine the logical order among changes made by Publisher entities to the same instance of data (i.e., matching Topic and key). **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucEnumerationParamDef		
Range	KEEP_ALL	On the publishing side, the Service will attempt to keep all samples (representing each value written) of each instance of data (identified by its key) managed by the DataWriter until they can be delivered to all subscribers. On the subscribing side, the Service will attempt to keep all samples of each instance of data (identified by its key) managed by the DataReader. These samples are kept until the application "takes" them from the Service via the take operation. **Tags:** atp.Status=draft	
	KEEP_LAST	On the publishing side, the Service will only attempt to keep the most recent "depth" samples of each instance of data (identified by its key) managed by the DataWriter. On the subscribing side, the DataReader will only attempt to keep the most recent "depth" samples received for each instance (identified by its key) until the application "takes" them via the DataReader's take operation. **Tags:** atp.Status=draft	
Default value	KEEP_LAST		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

SWS Item	[ECUC_Dds_00063]		
Parameter Name	DdsHistoryOrderDepth		
Parent Container	DdsHistory		
Description	Specifies the numbers of samples to keep. If DdsHistoryKind = KEEP_ALL, this parameter has not effect. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucIntegerParamDef		
Range	0 .. 65534		
Default value	1		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

No Included Containers

10.2.3.3.19 DdsResourceLimits

SWS Item			
Container Name	DdsResourceLimits		
Parent Container	DdsDataReaderQoS, DdsDataWriterQoS, DdsTopicQoS		
Description	It describes the DDS RESOURCE_LIMITS QoS policy. This QoS specifies the resources that the Service can consume in order to meet the requested QoS. This policy can be enabled only for Topics, DataReaders and DataWriters. **Tags:** atp.Status=draft		
Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

SWS Item	[ECUC_Dds_00064]		
Parameter Name	DdsResouceLimitsMaxInstances		
Parent Container	DdsResourceLimits		
Description	It represents the maximum number of instances DataWriter (or DataReader) can manage. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucIntegerParamDef		
Range	0 .. 65534		
Default value	65534		
Post-Build Variant Value	false		

▽

△

Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

SWS Item	**[ECUC_Dds_00062]**		
Parameter Name	DdsResouceLimitsMaxSamples		
Parent Container	DdsResourceLimits		
Description	It specifies the maximum number of datasamples the DataWriter (or DataReader) can manage across all the instances associated with it. Represents the maximum samples the middleware can store for any one DataWriter (or DataReader). It is inconsistent for this value to be less than max_samples_per_instance. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucIntegerParamDef		
Range	0 .. 65534		
Default value	65534		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

SWS Item	**[ECUC_Dds_00065]**		
Parameter Name	DdsResouceLimitsMaxSamplesPerInstance		
Parent Container	DdsResourceLimits		
Description	It represents the maximum number of samples of any one instance a DataWriter (or DataReader) can manage. It is inconsistent for this value to be greater than max_samples. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucIntegerParamDef		
Range	0 .. 65534		
Default value	65534		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

No Included Containers

10.2.3.3.20 DdsEntityFactory

SWS Item			
Container Name	DdsEntityFactory		
Parent Container	DdsDomainParticipantQoS, DdsPublisherQoS, DdsSubscriberQoS		
Description	It describes the DDS ENTITY_FACTORY QoS policy. This QoS controls the behavior of the entity when acting as a factory for other entities. In other words, configures the side-effects of the create and delete operations. This policy can be enabled only for DomainParticipantFactory, DomainParticipant, Publisher and Subscriber. **Tags:** atp.Status=draft		
Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

SWS Item	**[ECUC_Dds_00073]**		
Parameter Name	DdsEntityFactoryAutoenableCreatedEntities		
Parent Container	DdsEntityFactory		
Description	It specifies whether the entity acting as a factory automatically enables the instances it creates. If autoenable_created_entities==TRUE the factory will automatically enable each created Entity otherwise it will not. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucBooleanParamDef		
Default value	true		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: local		

No Included Containers

10.2.3.3.21 DdsWriterDataLifecycle

SWS Item	**[ECUC_Dds_00066]**
Container Name	DdsWriterDataLifecycle
Parent Container	DdsDataWriterQoS
Description	It describes the DDS WRITER_DATA_LIFECYCLE QoS policy. It specifies the behavior of the DataWriter with regards to the lifecycle of the datainstances it manages. This policy can be enabled only for DataWriter. **Tags:** atp.Status=draft

▽

△

Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

SWS Item	**[ECUC_Dds_00067]**		
Parameter Name	DdsAutodisposeUnregisteredInstances		
Parent Container	DdsWriterDataLifecycle		
Description	It controls whether a DataWriter will automatically dispose instances each time they are unregistered. The setting autodispose_unregistered_instances = TRUE indicates that unregistered instances will also be considered disposed. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucBooleanParamDef		
Default value	true		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: local		

No Included Containers

10.2.3.3.22 DdsReaderDataLifecycle

SWS Item	**[ECUC_Dds_00095]**		
Container Name	DdsReaderDataLifecycle		
Parent Container	DdsDataReaderQoS		
Description	It describes the DDS READER_DATA_LIFECYCLE QoS policy. It specifies the behavior of the DataReader with regards to the lifecycle of the data-instances it manages. This policy can be enabled only for DataReader. **Tags:** atp.Status=draft		
Post-Build Variant Multiplicity	false		
Multiplicity Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Configuration Parameters			

SWS Item	**[ECUC_Dds_00097]**
Parameter Name	DdsAutopurgeDisposedSamplesDelay
Parent Container	DdsReaderDataLifecycle

▽

△

Description	It indicates the duration the DataReader must retain information regarding instances that have the instance_state NOT_ALIVE_DISPOSED. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucIntegerParamDef		
Range	0 .. 65534		
Default value	65534		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

SWS Item	**[ECUC_Dds_00096]**		
Parameter Name	DdsAutopurgeNowriterSamplesDelay		
Parent Container	DdsReaderDataLifecycle		
Description	It indicates the duration the DataReader must retain information regarding instances that have the instance_state NOT_ALIVE_NO_WRITERS. **Tags:** atp.Status=draft		
Multiplicity	1		
Type	EcucIntegerParamDef		
Range	0 .. 65534		
Default value	65534		
Post-Build Variant Value	false		
Value Configuration Class	**Pre-compile time**	X	All Variants
	Link time	–	
	Post-build time	–	
Scope / Dependency	scope: ECU		

No Included Containers

10.3 Published Information

For details refer to the chapter 10.3 “Published Information” in SWS_BSWGeneral.

A Not applicable requirements

Na