Document Title	Requirements on Data Distribution Service
Document Owner	AUTOSAR
Document Responsibility	AUTOSAR
Document Identification No	1068

Document Status	published
Part of AUTOSAR Standard	Foundation
Part of Standard Release	R25-11

Document Change History

Date	Release	Changed by	Description
2025-11-27	R25-11	AUTOSAR Release Management	• Added Service Oriented Architecture requirements • Added Dynamic Discovery requirements
2024-11-27	R24-11	AUTOSAR Release Management	• Editorial changes
2023-11-23	R23-11	AUTOSAR Release Management	• Editorial changes
2022-11-24	R22-11	AUTOSAR Release Management	• Initial release

Disclaimer

Table of Contents

1 Scope of Document

This document specifies the requirements of the **Data Distribution Service (DDS)** [1], which shall be supported by AUTOSAR CP and AP.

DDS is a middleware protocol and API standard for data-centric connectivity standardized by the Object Management Group (OMG), so every requirement would be directly derived from the OMG standard.

A detailed list can be found on 4.2

2 Conventions to be used

2.1 Document Conventions

The representation of requirements in AUTOSAR documents follows the table specified in [TPS_STDT_00078], see [2, Standardization Template].

The verbal forms for the expression of obligation specified in [TPS_STDT_00053] shall be used to indicate requirements, see [2, Standardization Template].

In requirements, the following specific semantics shall be used (based on the Internet Engineering Task Force IETF).

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as:

- **SHALL:** This word means that the definition is an absolute requirement of the specification.
- **SHALL NOT:** This phrase means that the definition is an absolute prohibition of the specification.
- **MUST:** This word means that the definition is an absolute requirement of the specification due to legal issues.
- **MUST NOT:** This phrase means that the definition is an absolute prohibition of the specification due to legal constraints.
- **SHOULD:** This word, or the adjective "RECOMMENDED", mean that there may exist valid reasons in particular circumstances to ignore a particular item, but the full implications must be understood and carefully weighed before choosing a different course.
- **SHOULD NOT:** This phrase, or the phrase "NOT RECOMMENDED" mean that there may exist valid reasons in particular circumstances when the particular behavior is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behavior described with this label.
- **MAY:** This word, or the adjective OPTIONAL, means that an item is truly optional. One vendor may choose to include the item because a particular marketplace requires it or because the vendor feels that it enhances the product while another vendor may omit the same item. An implementation, which does not include a particular option, MUST be prepared to interoperate with another implementation, which does include the option, though perhaps with reduced functionality. In the same vein an implementation, which does include a particular option, MUST be prepared to interoperate with another implementation, which does not include the option (except, of course, for the feature the option provides.)

2.2 Requirements Guideline

The functional requirements defined in this document have been grouped on the basis of their functionality. The groups detailed in the next chapters are:

- **General:** [4.2.1] Different kind of requirements related to the DDS OMG standard and to the introduction on AUTOSAR CP and AP architectures.
- **Security:** [4.2.2] Requirements required to guarantee some degrees of Security.
- **Safety:** [4.2.3] Requirements required to guarantee some degrees of Safety according ISO-26262 [3].
- **DDS-DCPS Configuration:** [4.2.4]. Requirements stating the complaince with the Data-Centric Publisher/Subscriber communication paradigm.
- **Requirements for unidirectional DDS communication:** [4.2.5]. Requirement related to the used DDS transport protocol.
- **DDSI-RTPS:** [4.2.6]. Requirements stating the compliance with OMG DDSI-RTPS protocol [4]
- **Requirements for Service Oriented Architecture:** [4.2.7]. Requirements stating the compliance with the Service Oriented Architecture definition in AUTOSAR.
- **Dynamic Discovery:** [4.2.8]. Requirements stating the compliance with OMG SPDP, SEDP protocols [4] and AUTOSAR Service Discovery [5]

2.2.1 Requirements quality

No content

2.2.2 Requirements identification

No content

3 Acronyms and abbreviations

3.1 Acronyms

For acronyms and abbreviations refer to [6, AUTOSAR glossary].

Acronym	Description
DDS-RPC	DDS Remote Procedure Call. In this document DDS Remote Procedure Call (or Remote Procedure Call simply) refers to the concepts expressed into the specification Remote Procedure Call over DDS [7]. It introduces the concept of DDS Services. Services provide the mechanisms required to define and implement methods that can be invoked remotely by DDS client applications using the building blocks of the DDS data-centric publish-subscribe middleware [1].

Table 3.1: DDS supported Communication paths

3.2 Abbreviations

None

4 Requirements Specification

This chapter describes requirements for integration of DDS in AUTOSAR.

4.1 Functional Overview

The **Data Distribution Service (DDS)** [1] is a middleware protocol and API standard from the Object Management Group (OMG). It integrates the components of a system together, providing low-latency data connectivity, extreme reliability, and a scalable architecture.

The DDS Middleware is a software layer that abstracts the Application from the details of the operating system, network transport, and low-level data formats. Underlying details like data wire format, discovery, connections, reliability, protocols, transport selection, Quality of Service (QoS), security, etc. are managed transparently by the middleware.

DDS provides QoS-managed data sharing. Applications communicate by publishing and subscribing to Topics identified by their Topic name. Subscriptions can specify time and content filters and get only a subset of the data being published on the Topic. Different DDS Domains can be defined in a system and each will operate in complete isolation, unless explicit cross-domain routing components are explicitly deployed.

DDS is uniquely data centric. Data centricity ensures that all messages include the contextual information an application needs to understand the data it receives. The essence of data centricity is that DDS knows what data it stores and controls how to share that data. Programmers using traditional message-centric middleware must write code that sends messages. In contrast, programmers using data-centric middleware write code that specifies how and when to share data. Rather than managing all this complexity in the application (user) code, DDS transparently implements managed, efficient, safe and secure data sharin on behalf of the Application's needs.

4.1.1 Global data space

DDS Applications rely on a Data Model that is independent from the platform used. This model defines the global data space and specifies how Publishers and Subscribers refer to portions of this space. The Data Model can be as simple as a set of unrelated data-structures, each identified by a Topic and a Data Type.

Topics are uniquely identified by a name within the global data space. In addition, a Data Type provides structural information needed to tell the middleware how to manipulate the data. The use of typed interfaces implies the need for a generation tool to translate type descriptions into appropriate interfaces and implementations that fill the gap between the typed interfaces, the generic middleware and the platform-specifics (e.g. programming language).

The following definitions maybe useful to better understand the meaning and the context of requirements in this document:

Entity: The base object class of DDS, almost all the others are specializations of this one.

Topic: Topics associate a Name, a Data Type, and a set of QoS policy values. In addition to the topic QoS, the QoS of the `DataWriter` associated with that Topic and the QoS of the `Publisher` associated to the `DataWriter` control the behavior on the `Publisher`'s side, while the corresponding Topic, `DataReader`, and `Subscriber` QoS control the behavior on the `Subscriber`'s side.

Publisher: This Entity is responsible for data distribution. It may publish data of different Data Types to different Topics.

DataWriter: The application must use one or more `DataWriter`s to communicate to a `Publisher` the existence and value updates of a certain Topic. When data object values have been communicated to the `Publisher` through the appropriate `DataWriter`, it is the `Publisher`'s responsibility to perform the distribution (the `Publisher` will do this according to its own QoS, or the QoS attached to the corresponding `DataWriter`).

Subscriber: This Entity is responsible for data reception. It may receive data of different data types from different Topics.

DataReader: The application must use one or more `DataReader`s to communicate to a `Subscriber` interest in value updates of certain Topic. When data object values have been communicated to the `Subscriber` through the appropriate `DataReader`, it is the `Subscriber`'s responsibility to notify and hand the data to the Application (the `Subscriber` will do this according to its own QoS, or the QoS attached to the corresponding `DataReader`).

Domain: Represents a communication plane where only the `Publisher`s and the `Subscriber`s attached to the same Domain may interact through `DomainParticipant` Entities.

Quality of Services (QoS): QoS (Quality of Service) is a general concept that is used to specify the behavior of an Entity. QoS is comprised of individual QoS policies. Specific values for one or more QoS policies affecting diverse Entities may be grouped in QoS profiles.

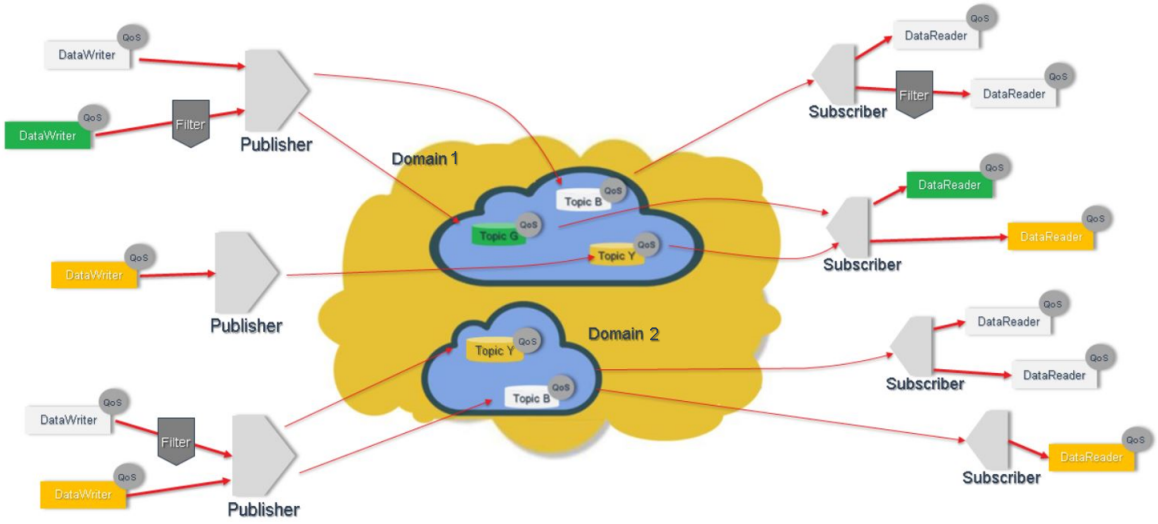

Figure 4.1: The DDS logical view

4.1.2 Service Oriented Architecture

In AUTOSAR a **Service Instance** is a functional entity implementing a **Service Interface**. Service Interfaces group named elements conforming to different kind of communication paradigms (mostly AUTOSAR specializations of general, well-known patterns such as, publish-subscribe and request-response).

In a Service-Oriented Architecture (SOA), applications can either provide (i.e. act as a server of) or require (i.e. act as a client of) Service Instances conforming to a pre-agreed Service Interface. In this kind of scenario the applications are loosely coupled and communicate over a middleware layer providing Service-oriented functionality.

DDS can realize these SOA functions by supporting implementation the five kinds of communication styles defined by the Communication Management Functional Cluster of the AUTOSAR Adaptive Platform [8]:

- **Discovery**: Ability of Service Instances to dynamically be instantiated and discovered in a network
- **Events**: Uni-directional typed data transmission from Service Providers to subscribing Service Consumers
- **Triggers**: Uni-directional trigger transmission from Service Providers to subscribing Service Consumers
- **Methods**: Bi- or uni-directional method invocation between Service Consumer and Service Providers
- **Fields**: Typed data elements on Service Providers that can either be explictly read, modified or subscribed to by Service Consumers

This document defines the general requirements for DDS Middleware implementations to realize this level of SOA support in AUTOSAR, similarly, and in parallel to, the SOME/IP protocol [9].

4.1.3 Dynamic Discovery

The following sub-sections describe the different layers of discovery protocols leading up to SOA Service Instance Discovery support under AUTOSAR and DDS.

4.1.3.1 OMG DDS Discovery

DDS Discovery is the behind-the-scenes technology in which `DomainParticipant`s, `DataWriter`s, and `DataReader`s on different ECUs dynamically learn about each other's existence and parameters.

Each `DomainParticipant` maintains an internal volatile database of information about all the known `DomainParticipant`s and the `DataReader`s/`DataWriter`s they host. This database is what makes it possible for `DataWriter`s and `DataReader`s to be matched and, ultimately, communicate with each other. To create and refresh the database, each application follows a well-known discovery process.

The RTPS specification splits up the discovery protocol into two independent protocols: **Participant Discovery Protocol** (PDP) and **Endpoint Discovery Protocol** (EDP).

A Participant Discovery Protocol specifies how Participants discover each other in the network.

Once two Participants have discovered each other, they exchange information on the Endpoints (`DataWriter`s and `DataReader`s) they contain using an Endpoint Discovery Protocol.

DDS Middleware implementations may choose to support multiple PDPs and EDPs, possibly vendor-specific. As long as two Participants have at least one PDP and EDP in common, they can exchange discovery information.

For the purpose of interoperability in the case of dynamic discovery, as specified in [FO_RS_Dds_00040] and [FO_RS_Dds_00041], all DDS implementations shall provide at least the following discovery protocols (see [4] for details):

- Simple Participant Discovery Protocol (SPDP)
- Simple Endpoints Discovery Protocol (SEDP)

4.1.3.2 AUTOSAR Service Instance Discovery

Once DDS `DomainParticipant`s have discovered each other, an additional set of exchanges takes place in order to:

- Advertising of availabe Service Instances by Applications providing them
- Discovery of, and binding to, Service Instances by Applications requiring them

For details about the AUTOSAR DDS Service Discovery Protocol refer to [5].

4.2 Functional Requirements

4.2.1 General

[FO_RS_Dds_00001] DDS Compliance ⌈

Description:	The DDS components of AUTOSAR shall comply with the DDS Minimum Profile defined in [1], the DDS Wire Interoperability protocol (DDSI-RTPS) defined in [4], and the DDS-XTYPES Minimal Programming Interface and Network Interoperability Profiles defined in [10].
Rationale:	Interoperability with other nodes implementing DDS
Dependencies:	–
Use Case:	• Intercommunications between different kinds of architectures (e.g: AP/CP communication compatibility) • Intercommunicate with simulators or ADAS systems based on DDS (e.g. based on ROS2)
AppliesTo:	FO, CP
Supporting Material:	OMG Data Distribution Service protocol specification [1] OMG DDS Interoperability Wire Protocol [4] OMG Extensible and Dynamic Topic Types for DDS [10]

⌋

[FO_RS_Dds_00002] DDS standard serialization rules ⌈

Description:	Data packets to be sent to the network shall be compliant with the DDS Interoperability Wire Protocol (DDSI-RTPS) defined in [4]
Rationale:	Interoperability with other nodes implementing DDS
Dependencies:	–
Use Case:	• Intercommunications between different kinds of architectures • Intercommunicate with simulators or ADAS systems based on DDS (e.g. based on ROS2)
Supporting Material:	OMG DDS Interoperability Wire Protocol [4]
AppliesTo:	FO, CP

⌋

[FO_RS_Dds_00004] DDS payload serialization rules ⌈

Description:	The serialization of the payload shall be done according to the DDS standard serialization rules defined in [10]
Rationale:	Interoperability with other nodes implementing DDS
Dependencies:	–
Use Case:	• Intercommunications between different kinds of architectures • Intercommunicate with simulators or ADAS systems based on DDS (e.g. based on ROS2)
Supporting Material:	OMG Extensible and Dynamic Topic Types for DDS [10]
AppliesTo:	FO, CP

⌋

[FO_RS_Dds_00005] DDS Quality of Service ⌈

Description:	The DDS components of AUTOSAR shall support DDS QoS policies
Rationale:	Leverage QoS policy features
Dependencies:	–
Use Case:	• Receive data at different rates • Time-sensitive Network Support • Support on Safety • Custom presentation order
Supporting Material:	OMG Data Distribution Service protocol specification [1]
AppliesTo:	FO, CP

⌋

[FO_RS_Dds_00006] The DDS AUTOSAR components receive unserialized data ⌈

Description:	In order to better integrate data-centric capabilities of DDS, the DDS components of AUTOSAR shall receive and produce data in unserialized format.
Rationale:	To exploit all DDS functionalities, the DDS components shall work with untransformed data. The DDS middleware shall be aware of application data type it receives.
Dependencies:	–
Use Case:	–
Supporting Material:	OMG Data Distribution Service protocol specification [1]
AppliesTo:	FO, CP, AP

⌋

[FO_RS_Dds_00007] Type Definition ⌈

Description:	The DDS middleware shall define an unequivocal mapping of AUTOSAR and DDS types relevant in DDS communication, for each supported AUTOSAR platform.
Rationale:	Application Software Components will deal exclusively with AUTOSAR APIs and type system, thus an internal mapping supporting data format consistency is in order.
Dependencies:	–
Use Case:	–
Supporting Material:	OMG Extensible and Dynamic Topic Types for DDS [10]
AppliesTo:	FO, CP

⌋

[FO_RS_Dds_00008] Customization ⌈

Description:	DDS shall allow customization of the DDS entities within the ECU.
Rationale:	Interoperate with other DDS-based systems, adapting to their data architecture and specific QoS policy.
Dependencies:	–
Use Case:	–
Supporting Material:	–
AppliesTo:	FO, CP

⌋

4.2.2 Security

[FO_RS_Dds_00009] Security mechanism ⌈

Description:	The DDS middleware shall suppport and implement the DDSI-RTPS wire protocol extensions of DDS Security [11] in order to guarantee data integrity and data authentication.
Rationale:	Because external and hybrid communication will be provided, the communication must be secured
Dependencies:	–
Use Case:	• Secure interoperability with other platforms in the vehicle
Supporting Material:	OMG DDS Security Specification [11]
AppliesTo:	FO, CP

⌋

4.2.3 Safety

[FO_RS_Dds_00010] Safety mechanism ⌈

Description:	The DDS middleware shall support ISO26262-compliant E2E [12] safety mechanisms, without breaking DDS and DDSI-RTPS specification compliance
Rationale:	Because external and hybrid communication will be provided, the communication must incorporate safety mechanisms
Dependencies:	–
Use Case:	• Safe interoperability with other platforms in the vehicle
Supporting Material:	ISO 26262 [3]
AppliesTo:	FO, CP

⌋

4.2.4 DDS-DCPS Configuration

[FO_RS_Dds_00015] Publish ⌈

Description:	The DDS AUTOSAR component shall permit to publish data on specific topics
Rationale:	Send data
Dependencies:	–
Use Case:	Send typed data
Supporting Material:	OMG Data Distribution Service protocol specification [1]
AppliesTo:	FO, CP

⌋

[FO_RS_Dds_00016] Subscribe ⌈

Description:	The DDS AUTOSAR component shall permit to subscribe to specific topics
Rationale:	Receive typed data
Dependencies:	–
Use Case:	Receive specific data type
Supporting Material:	OMG Data Distribution Service protocol specification [1]
AppliesTo:	FO, CP

⌋

4.2.5 Requirements for unidirectional DDS communication

[FO_RS_Dds_00017] Transport protocol ⌈

Description:	The AUTOSAR DDS component shall support different transport protocols underneath.
Rationale:	The AUTOSAR DDS component is unaware of the transport protocol.
Dependencies:	–
Use Case:	• DDS over UDP for time critical communications. • DDS over TCP for time-uncritical communication of large data. • Shared-memory for intra-ecu communications.
Supporting Material:	DDS Interoperability Wire Protocol ([4], chapter 9)
AppliesTo:	FO, CP

⌋

4.2.6 Requirements for DDSI-RTPS

[FO_RS_Dds_00019] RTPS message encapsulation ⌈

Description:	The DDS components of AUTOSAR shall encapsulate data into RTPS messages
Rationale:	DDS intercommunicate by using RTPS messages
Dependencies:	–
Use Case:	DDS intercommunication
Supporting Material:	–
AppliesTo:	FO, CP

⌋

[FO_RS_Dds_00020] RTPS message decapsulation ⌈

Description:	The DDS components of AUTOSAR shall decapsulate data from RTPS messages
Rationale:	DDS intercommunicate by using RTPS messages
Dependencies:	–
Use Case:	DDS intercommunication
Supporting Material:	–
AppliesTo:	FO, CP

⌋

4.2.7 Requirements for SOA

[FO_RS_Dds_00030] DDS Event communication support ⌈

Description:	DDS shall support event communication, which is a uni-directional communication that is produced and sent by the service provider.
Rationale:	Event based communication needs to be considered in the communication over the network
Dependencies:	–
Use Case:	Communication of data which are produced on event-based, e.g gear-shifts
Supporting Material:	Requirements on SOME/IP Protocol
AppliesTo:	FO, CP

⌋

[FO_RS_Dds_00031] DDS Method communication support ⌈

Description:	DDS must support method communication, which includes bidirectional and unidirectional RPC communication. In the latter case, the callee is not required to inform the caller of the result. In the former case, however, the caller must inform the callee of the result.
Rationale:	A remote procedure call either needs receive the result or doesn't need it.
Dependencies:	–
Use Case:	Remote procedure call
Supporting Material:	Requirements on SOME/IP Protocol, OMG Remote Procedure Call over DDS
AppliesTo:	FO, CP

⌋

[FO_RS_Dds_00032] DDS error handling of bi-directional method communication support ⌈

Description:	DDS shall support error handling of bi-directional method communication and communicate errors on callee side back to the caller.
Rationale:	For properties held at a central party, this communication pattern provides the explicit access and subscription-oriented access to data.
Dependencies:	–
Use Case:	One party in the vehicle holds a central property which is set and/or used by multiple other parties.
Supporting Material:	Requirements on SOME/IP Protocol, OMG Remote Procedure Call over DDS
AppliesTo:	FO, CP

⌋

[FO_RS_Dds_00033] DDS shall support field communication with getters, setters and notification events. ⌈

Description:	DDS shall support field communication with getters, setters and notification events.
Rationale:	For properties held at a central party, this communication pattern provides the explicit access and subscription-oriented access to data.
Dependencies:	–
Use Case:	One party in the vehicle holds a central property which is set and/or used by multiple other parties.
Supporting Material:	Requirements on SOME/IP Protocol, OMG Remote Procedure Call over DDS
AppliesTo:	FO, CP

⌋

4.2.8 Requirements for Dynamic Discovery

[FO_RS_Dds_00040] DDS shall optionally support SPDP according to DDS-RTPS specification ⌈

Description:	DDS shall optionally support Simple Participant Discovery Protocol (SPDP). NOTE: SPDP needs to be supported if a static remote Domain participant configuration is missing.
Rationale:	This protocol represents the only standard implementation of a generic Simple Participant Protocol (PDP). It represents the first step of whole DDS Dynamic Discovery procedure. Goal of each participant is to achieve a complete list of all remote participant that are in its peers list. The peers list is the list of nodes with which a participant may communicate.
Dependencies:	–
Use Case:	DDS middleware application
Supporting Material:	OMG DDS Interoperability Wire Protocol (DDS-RTPS)
AppliesTo:	FO, CP

⌋

[FO_RS_Dds_00041] DDS shall optionally support SEDP according to DDS-RTPS specification ⌈

Description:	DDS shall optionally support Simple Endpoints Discovery Protocol (SEDP). NOTE: SEDP needs to be supported if a static remote Endpoints(writers, readers) configuration is missing.

▽

△

Rationale:	This protocol represents the only standard implementation of a generic Simple Endpoints Protocol (EDP). It represents the second step of whole DDS Dynamic Discovery procedure. Once the list of remote participants is known, goal for each of them is to achieve a complete list of all owned entities (i.e RTPS writer/reader)
Dependencies:	–
Use Case:	DDS middleware application
Supporting Material:	OMG DDS Interoperability Wire Protocol (DDS-RTPS)
AppliesTo:	FO, CP

⌋

[FO_RS_Dds_00042] DDS shall support the Domain Participant USER_DATA QoS policy, if Service Discovery as described by AUTOSAR DDS Service Discovery Protocol specification is to be supported ⌈

Description:	DDS shall support the Domain Participant USER_DATA QoS policy, if Service Discovery as described by FO PRS DDSServiceDiscoveryProtocol is to be supported
Rationale:	USER_DATA QoS policy is necessary to be compliant with AP
Dependencies:	–
Use Case:	DDS middleware in AUTOSAR application
Supporting Material:	OMG DDS Interoperability Wire Protocol (DDS-RTPS), AUTOSAR Service Discovery Protocol specification
AppliesTo:	FO, CP

⌋

[FO_RS_Dds_00043] DDS shall support the Publisher/Subscriper PARTITION QoS policy, if Partition-based Resource Identification as described by AUTOSAR DDS Service Discovery Protocol specification is to be supported ⌈

Description:	DDS shall support the Publisher/Subscriper PARTITION QoS policy, if Partition-based Resource Identification as described by FO PRS DDSServiceDiscoveryProtocol is to be supported
Rationale:	The Partition-based is the basic Resource Identification Mechanisms defining how Service Interfaces and their individual Instances (the 'Resources') are uniquely instantiated and addressable with a particular DDS Domain. In this mechanism DDS Publisher and Subscriber Entity PARTITION QoS policy is leveraged to isolate each Service Instance and their consumers into a uniquely named DDS Partition (refer to FO PRS DDSServiceDiscoveryProtocol)
Dependencies:	–
Use Case:	DDS middleware in AUTOSAR application
Supporting Material:	OMG DDS Interoperability Wire Protocol (DDS-RTPS), AUTOSAR Service Discovery Protocol specification

▽

△

AppliesTo:	FO, CP

⌋

4.3 Non-Functional Requirements

None.

5 References

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

[2] Standardization Template
AUTOSAR_FO_TPS_StandardizationTemplate

[3] ISO 26262:2018 Road vehicles -– Functional Safety
https://www.iso.org

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

[5] Specification of DDS Service Discovery Protocol
AUTOSAR_FO_PRS_DDSServiceDiscoveryProtocol

[6] Glossary
AUTOSAR_FO_TR_Glossary

[7] RPC over DDS, Version 1.0
https://www.omg.org/spec/DDS-RPC/1.0

[8] Specification of Communication Management
AUTOSAR_AP_SWS_CommunicationManagement

[9] SOME/IP Protocol Specification
AUTOSAR_FO_PRS_SOMEIPProtocol

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

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

[12] Specification of SW-C End-to-End Communication Protection Library
AUTOSAR_CP_SWS_E2ELibrary

A Appendix

No content

B Change history of AUTOSAR traceable items

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

B.1 Traceable item history of this document according to AUTOSAR Release R25-11

B.1.1 Added Requirements in R25-11

Number	Heading
[FO_RS_Dds_00030]	DDS Event communication support
[FO_RS_Dds_00031]	DDS Method communication support
[FO_RS_Dds_00032]	DDS error handling of bi-directional method communication support
[FO_RS_Dds_00033]	DDS shall support field communication with getters, setters and notification events.
[FO_RS_Dds_00040]	DDS shall optionally support SPDP according to DDS-RTPS specification
[FO_RS_Dds_00041]	DDS shall optionally support SEDP according to DDS-RTPS specification
[FO_RS_Dds_00042]	DDS shall support the Domain Participant USER_DATA QoS policy, if Service Discovery as described by AUTOSAR DDS Service Discovery Protocol specification is to be supported
[FO_RS_Dds_00043]	DDS shall support the Publisher/Subscriper PARTITION QoS policy, if Partition-based Resource Identification as described by AUTOSAR DDS Service Discovery Protocol specification is to be supported

Table B.1: Added Requirements in R25-11

B.1.2 Changed Requirements in R25-11

none

B.1.3 Deleted Requirements in R25-11

none

B.2 Traceable item history of this document according to AUTOSAR Release R24-11

B.2.1 Added Requirements in R24-11

none

B.2.2 Changed Requirements in R24-11

Number	Heading
[FO_RS_Dds_00001]	DDS Compliance
[FO_RS_Dds_00002]	DDS standard serialization rules
[FO_RS_Dds_00004]	DDS payload serialization rules
[FO_RS_Dds_00005]	DDS Quality of Service
[FO_RS_Dds_00006]	The DDS AUTOSAR components receive unserialized data
[FO_RS_Dds_00007]	Type Definition
[FO_RS_Dds_00008]	Customization
[FO_RS_Dds_00009]	Security mechanism
[FO_RS_Dds_00010]	Safety mechanism
[FO_RS_Dds_00015]	Publish
[FO_RS_Dds_00016]	Subscribe
[FO_RS_Dds_00017]	Transport protocol
[FO_RS_Dds_00019]	RTPS message encapsulation
[FO_RS_Dds_00020]	RTPS message decapsulation

Table B.2: Changed Requirements in R24-11

B.2.3 Deleted Requirements in R24-11

none

B.3 Traceable item history of this document according to AUTOSAR Release R23-11

B.3.1 Added Requirements in R23-11

Number	Heading
[FO_RS_Dds_00001]	DDS Compliance
[FO_RS_Dds_00002]	DDS standard serialization rules
[FO_RS_Dds_00004]	DDS payload serialization rules
[FO_RS_Dds_00005]	DDS Quality of Service
[FO_RS_Dds_00006]	The DDS AUTOSAR components receive unserialized data
[FO_RS_Dds_00007]	Type Definition
[FO_RS_Dds_00008]	Customization
[FO_RS_Dds_00009]	Security mechanism
[FO_RS_Dds_00010]	Safety mechanism
[FO_RS_Dds_00015]	Publish
[FO_RS_Dds_00016]	Subscribe
[FO_RS_Dds_00017]	Transport protocol
[FO_RS_Dds_00019]	RTPS message encapsulation
[FO_RS_Dds_00020]	RTPS message decapsulation

Table B.3: Added Requirements in R23-11

B.3.2 Changed Requirements in R23-11

none

B.3.3 Deleted Requirements in R23-11

none

B.4 Traceable item history of this document according to AUTOSAR Release R22-11

B.4.1 Added Requirements in R22-11

Number	Heading
[FO_RS_Dds_00001]	DDS Compliance
[FO_RS_Dds_00002]	DDS standard serialization rules
[FO_RS_Dds_00004]	DDS payload serialization rules
[FO_RS_Dds_00005]	DDS Quality of Service
[FO_RS_Dds_00007]	Type Definition
[FO_RS_Dds_00008]	Customization
[FO_RS_Dds_00009]	Security mechanism
[FO_RS_Dds_00010]	Safety mechanism
[FO_RS_Dds_00015]	Publish
[FO_RS_Dds_00016]	Subscribe
[FO_RS_Dds_00017]	Transport protocol
[FO_RS_Dds_00019]	RTPS message encapsulation
[FO_RS_Dds_00020]	RTPS message decapsulation
[FO_RS_Dds_CONSTR_00006]	The DDS AUTOSAR components receive unserialized data

Table B.4: Added Requirements in R22-11

B.4.2 Changed Requirements in R22-11

none

B.4.3 Deleted Requirements in R22-11

none