Document Title	Integration of DDS Security
Document Owner	AUTOSAR
Document Responsibility	AUTOSAR
Document Identification No	1027

Document Status	published
Part of AUTOSAR Standard	Adaptive Platform
Part of Standard Release	R25-11

Document Change History

Date	Release	Changed by	Description
2025-11-27	R25-11	AUTOSAR Release Management	• No content changes
2024-11-27	R24-11	AUTOSAR Release Management	• No content changes
2023-11-23	R23-11	AUTOSAR Release Management	• ID format updated, model references fixed ([TR_DDSS_00204], [TR_DDSS_00005], [TR_DDSS_00104], [TR_DDSS_00202], [TR_DDSS_00205])
2022-11-24	R22-11	AUTOSAR Release Management	• No content changes
2021-11-25	R21-11	AUTOSAR Release Management	• Initial release

Disclaimer

Table of Contents

1 Introduction

This Technical Report provides additional information to the DDS Network Binding of the Communications Management functional cluster of the AUTOSAR Adaptive Platform, as defined by [1].

DDS Security, as defined in [2], is a complementary standard to DDS, providing transport-independent security measures (authentication, secrecy, non-repudiation, integrity, access control and logging) without requiring changes to application logic.

1.1 Objectives

This document aims at mapping DDS Service Interface and Instance Deployment models, as well as IAM Communications Grant models, to DDS QoS policies, and DDS Security certificate, governance and permission documents as defined by [2].

1.2 Scope

This document builds on the DDS Network Binding as specified by [1] and supports, in summary, the following security mechanisms:

- Per-instance, per-event access control, along with secrecy and authentication configuration for in-band and out-of-band traffic
- Per-instance, per-field notifier access control, along with secrecy and authentication configuration for in-band and out-of-band traffic
- Per instance methods access control along with secrecy and authentication configuration for in-band and out-of-band traffic
- Per instance field methods (`Get`/`Set`) access control along with secrecy and authentication configuration for in-band and out-of-band traffic

As noted above, fine-grained security controls for independent methods and field methods (`Get`/`Set`) are not supported by DDS Security at the moment, due to the specific design of the DDS Network Binding, where all methods belonging to a single Service Interface Instance are multiplexed over a limited set of DDS Topics.

2 Definition of terms and acronyms

2.1 Acronyms and abbreviations

Abbreviation / Acronym:	Description:
ACL	Access Control List
CA	Certificate Authority
DDS	Data Distribution Service
IAM	Identity and Access Management
QoS	Quality of Service
URI	Uniform Resource Identifier

2.2 Definition of terms

Not applicable.

3 Related Documentation

3.1 Input documents & related standards and norms

[1] Specification of Communication Management
AUTOSAR_AP_SWS_CommunicationManagement

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

[3] Specification of Manifest
AUTOSAR_AP_TPS_ManifestSpecification

[4] Specification of Execution Management
AUTOSAR_AP_SWS_ExecutionManagement

4 AUTOSAR Metamodel to DDS Security mappings

4.1 Configuration workflow

Integrators should not manually manipulate DDS Security artifacts, but rather update related the AUTOSAR design elements, then re-generate and re-deploy the DDS Security artifacts:

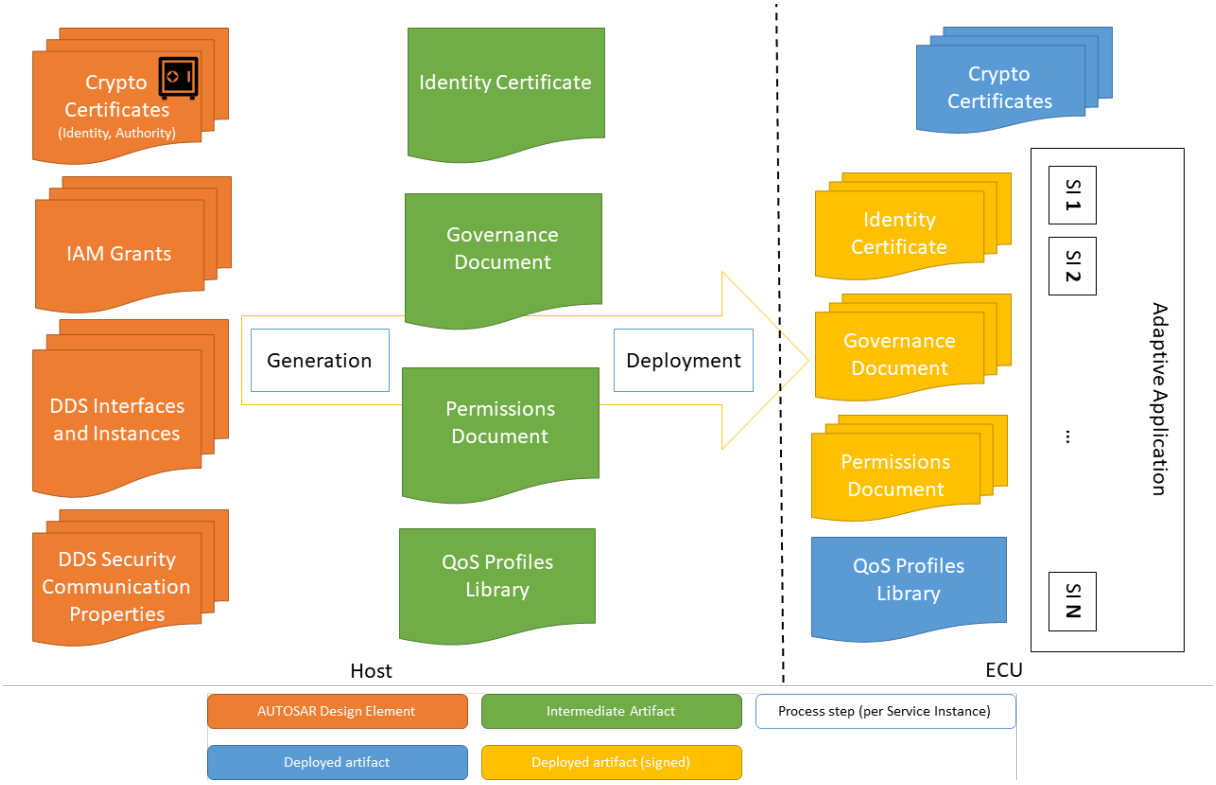

Figure 4.1: Workflow for DDS Security artifact generation and deployment

Although the following sections describe this process in detail, a brief summary is presented here for clarity and ease of understanding:

1. DDS-specific deployment for Service Interfaces and Service Instances is modelled as prescribed in [3], including DDS Security Communication Properties (`DdsSecureComProps`) and the cryptographic resources associated to them (`CryptoCertificate`)
2. Following the detailed procedures shown in the next sections, a set of intermediate DDS Security-specific artifacts are produced for each Provided or Required DDS Service Instance, portraying modelled instance identity, domain governance policies, participant policies and QoS policies
3. During deployment, for each service instance, identity certificates, governance and permission documents are signed using secret key material by the host, and deployed alongside relevant crypto certificates (without the private key part) and the QoS profiles library

4. In run-time, Adaptive Applications load the instance certificates, governance and permission documents referenced by the QoS profile assigned to each service instance in the QoS Profiles Library. Deployed crypto certificates (holding no secret key material at all, only public keys) are used to verify signatures for both own and foreign identity, governance and permission documents

4.2 Provisioning of DDS Security artifacts

[TR_DDSS_00001] Artifacts required by Provided or Required Service Instances

Status: DRAFT

⌈For each `DdsServiceInstanceToMachineMapping` referencing a `DdsSecureComProps` object, the following artifacts shall be uniquely generated and deployed for access by the host `Process` during runtime along with the processed manifest:

- A unique, CA-signed DDS Security Governance Document, with contents according to [TR_DDSS_00101]
- A unique, CA-signed DDS Security Permissions Document, with contents according to [TR_DDSS_00201]
- A QoS profile to be referenced from `DdsProvidedServiceInstance` or `DdsRequiredServiceInstance` via `qosProfile`, with Domain Participant QoS properties set according to [TR_DDSS_00002], [TR_DDSS_00003], [TR_DDSS_00004], [TR_DDSS_00005], [TR_DDSS_00006] and [TR_DDSS_00007]

⌋

[TR_DDSS_00002] Identity Certificate Authority

Status: DRAFT

⌈The `dds.sec.auth.identity_ca` property shall be set to the short name path of the `CryptoCertificate` referenced by the `identityCertificateAuthority` attribute via `governance`, or an URI referencing a `CryptoCertificate` rendition that's supported by the DDS Security implementation (e.g. `file://...`).⌋

[TR_DDSS_00003] Identity Certificate

Status: DRAFT

⌈The `dds.sec.auth.identity_certificate` property shall be set to the short name path of the `CryptoCertificate` referenced by `identity`, or an URI referencing a `CryptoCertificate` rendition that's supported by the DDS Security implementation (e.g. `file://...`).⌋

[TR_DDSS_00004] Private Key

Status: DRAFT

⌈The `dds.sec.auth.private_key` property shall be set to the short name path of the `CryptoKeySlot` referenced, via `CryptoCertificate-ToCryptoKeySlotMapping`, by the `CryptoCertificate` defined in the `dds.sec.auth.identity_certificate` property, or an URI referencing a `CryptoKeySlot` rendition that's supported by the DDS Security implementation (e.g. `file://...`).⌋

[TR_DDSS_00005] Permissions Certificate Authority

Status: DRAFT

⌈The `dds.sec.auth.permissions_ca` property shall be set to the short name path of the `CryptoCertificate` referenced by the `permissionCertificateAuthority` attribute via `governance`, or an URI referencing a `CryptoCertificate` rendition that's supported by the DDS Security implementation (e.g. `file://...`).⌋

[TR_DDSS_00006] Governance Document

Status: DRAFT

⌈The `dds.sec.access.governance` property shall be set to the short name path or URI of the CA-signed DDS Security Governance Document created in the context of [TR_DDSS_00001].⌋

[TR_DDSS_00007] Permissions Document

Status: DRAFT

⌈The `dds.sec.access.permissions` property shall be set to the short name path or URI of the CA-signed DDS Security Permissions Document created in the context of [TR_DDSS_00001].⌋

The dual nature (short name paths or URIs) of these properties allows sensitive crypto resources and related documents to be addressed from sources of various kinds, such as filesystems (e.g. `file://...`) or AUTOSAR CryptoAPI key slot specifiers (e.g. `/CryptoCertiticates/Identity`).

4.3 Provisioning of the DDS Security Governance Document

In DDS Security, all Domain Participants communicating in the same secure domain operate under an authentic set of governance rules described in governance documents modelled via `DdsSecureGovernance`.

[TR_DDSS_00101] Governance Document

Status: DRAFT

⌈In the DDS Security Governance Document associated to each Service Instance through `governance` via `secureComPropsForDds` in the context of

[TR_DDSS_00001], a domain_rule element shall be incorporated under the domain_access_rules element as follows:

- The allow_unauthenticated_participants element is set to the value of allowUnauthenticatedParticipants (via governance)
- The enable_join_access_control element is set to the value of enableJoinAccessControl (via governance)
- The discovery_protection_kind element is set to the value of discoveryProtectionKind (via governance)
- The liveliness_protection_kind element is set to the value of livelinessProtectionKind (via governance)
- The rtps_protection_kind element is set to the value of rtpsProtectionKind (via governance)
- One topic_access_rules element as described by [TR_DDSS_00102], [TR_DDSS_00103] and [TR_DDSS_00104]

⌋

[TR_DDSS_00102] Generic topic access rules

Status: DRAFT

⌈At least one single "catch-all" topic access rule with topic expression ara.com://services/* shall be added under the topic_access_rules element of the domain_rule element defined by [TR_DDSS_00101]. Finer-grained sets of topic access rules (e.g., per Service Interface or Service Interface element) are acceptable as long as they follow rules expressed by [TR_DDSS_00103] and [TR_DDSS_00104].⌋

[TR_DDSS_00103] Detailed topic access rules Service Discovery

Status: DRAFT

⌈One single topic access rule with topic expression ara.com://services/discovery shall be added under the topic_access_rules element of the domain_rule element defined by [TR_DDSS_00101]. Specific access parameters for this topic are implementation dependent.⌋

[TR_DDSS_00104] Detailed topic access rules for Service Interfaces

Status: DRAFT

⌈For each DdsServiceInstanceToMachineMapping referencing a DdsSecureComProps object, each associated DdsServiceInterfaceDeployment may extend the associated (in the context of [TR_DDSS_00101]) Governance Document topic_access_rules element with topic_rule elements as follows:

- Add one topic_rule element for each DdsEventDeployment associated to the DdsServiceInterfaceDeployment, with a set of sub-elements mirroring the TopicAccessRule values referenced by eventTopicAccessRule, and

a topic_expression sub-element set to ara.com://services/<ServiceInterface>/*/<EventTopicName>, where:

  - <ServiceInterface> takes the value of serviceInterfaceId
  - <EventTopicName> takes the value of topicName

- Add one topic_rule element, similar to the aforementioned DdsEventDeployment element, for each DdsFieldDeployment referencing a field with hasNotifier set to True via field
- Add two topic_rule elements, each with a set of sub-elements mirroring the TopicAccessRule referenced by methodTopicsAccessRule, and topic_expression sub-elements respectively set to ara.com://services/<ServiceInterface>/*/<MethodRequestTopicName> and ara.com://services/<ServiceInterface>/*/<MethodReplyTopicName>, where:
  - <ServiceInterface> takes the value of serviceInterfaceId
  - <MethodRequestTopicName> takes the value of methodRequestTopicName
  - <MethodReplyTopicName> takes the value of methodReplyTopicName
- Add two topic_rule elements, each with a set of sub-elements mirroring the TopicAccessRule referenced by fieldTopicsAccessRule, and topic_expression sub-elements respectively set to ara.com://services/<ServiceInterface>/*/<FieldRequestTopicName> and ara.com://services/<ServiceInterface>/*/<FieldReplyTopicName>, where:
  - <ServiceInterface> takes the value of serviceInterfaceId
  - <FieldRequestTopicName> takes the value of fieldRequestTopicName
  - <FieldReplyTopicName> takes the value of fieldReplyTopicName

⌋

4.4 Provisioning of the DDS Security Permissions Document

In DDS Security, all Domain Participants communicating in the same secure domain operate under an authentic set of ACL-like policies applicable to domains, partitions, topics and topic instances, described in permissions documents modelled via ComGrants.

[TR_DDSS_00201] Permissions file contents for DDS IAM Remote Subjects

Status: DRAFT

⌈In the DDS Security Permissions Document associated to each Service Instance via `secureComPropsForDds`in the context of [TR_DDSS_00001], a `grant` element shall added under the `permissions` element, including:

- A `subject_name` element set to the subject name field of the certificate referenced by `identity`.
- An `allow_rule` element, including:
  - A `domains` element mirroring `domainId` through `governance`
  - A `publish` element with contents for provided and required service instances according to [TR_DDSS_00202] and [TR_DDSS_00204], respectively
  - A `subscribe` element with contents for provided and required service instances according to [TR_DDSS_00203] and [TR_DDSS_00205], respectively
- A `default` element set to `DENY`

⌋

[TR_DDSS_00202] Allow/publish rules for Provided Service Instances

Status: DRAFT

⌈For each `DdsServiceInstanceToMachineMapping` referencing a `DdsSecureComProps` object, each associated `DdsProvidedServiceInstance` shall extend the associated (in the context of [TR_DDSS_00201]) Permissions Document `publish` element as follows:

- Under the `partitions` element:
  - Add, if it doesn't exist yet, an empty `partition` element (for updating the discovery topic)
  - Add an additional `partition` element with value `ara.com://services/<ServiceInterface>/<ServiceInstance>`, where:
    - `ServiceInterface` takes the value of `serviceInterfaceId` (through `serviceInterfaceDeployment`)
    - `ServiceInstance` takes the value of `serviceInstanceId`
- Under the `topics` element:
  - Add, if it doesn't exist yet, a `topic` element with value `ara.com://services/discovery` (for updating the discovery topic)
  - Add two `topic` elements for each `ComEventGrant` referencing the current `DdsProvidedServiceInstance` via `serviceInstance`

with values ara.com://services/<ServiceInterface>/<ServiceInstance>/<EventTopicName> and ara.com://services/<ServiceInterface>/<Major>.<Minor>/<EventTopicName> where:

  * ServiceInterface takes the value of serviceInterfaceId (through serviceInterfaceDeployment)
  * ServiceInstance takes the value of serviceInstanceId
  * Major and Minor takes the value of majorVersion and minorVersion (via serviceInterfaceDeployment)
  * EventTopicName takes the value of topicName (through serviceDeployment)

- Add two topic elements, similar to the aforementioned ComEventGrant elements, for each ComFieldGrant referencing a field with hasNotifier set to True via serviceDeployment
- Add four topic elements with values ara.com://services/<ServiceInterface>/<ServiceInstance>/<MethodsTopicName>, ara.com://services/<ServiceInterface>/<Major>.<Minor>/<MethodsTopicName>, ara.com://services/<ServiceInterface>/<ServiceInstance>/<FieldsTopicName>, ara.com://services/<ServiceInterface>/<Major>.<Minor>/<FieldsTopicName> where:

  * ServiceInterface takes the value of serviceInterfaceId (through serviceInterfaceDeployment)
  * ServiceInstance takes the value of serviceInstanceId
  * Major and Minor takes the value of majorVersion and minorVersion (via serviceInterfaceDeployment)
  * MethodsTopicName takes the value of methodReplyTopicName (through serviceInterfaceDeployment)
  * FieldsTopicName takes the value of fieldReplyTopicName (through serviceInterfaceDeployment)

⌋

[TR_DDSS_00203] Allow/subscribe rules for Provided Service Instances

Status: DRAFT

⌈For each DdsServiceInstanceToMachineMapping referencing a DdsSecureComProps object, each associated DdsProvidedServiceInstance shall extend the associated (in the context of [TR_DDSS_00201]) Permissions Document subscribe element as follows:

- Under the partitions element:

  - Add a `partition` element with value `ara.com://services/<ServiceInterface>/<ServiceInstance>`, where:
    * `ServiceInterface` takes the value of `serviceInterfaceId` (through `serviceInterfaceDeployment`)
    * `ServiceInstance` takes the value of `serviceInstanceId`
- Under the `topics` element:
  - Add four `topic` elements with values `ara.com://services/<ServiceInterface>/<ServiceInstance>/<MethodsTopicName>`, `ara.com://services/<ServiceInterface>/<Major>.<Minor>/<MethodsTopicName>`, `ara.com://services/<ServiceInterface>/<ServiceInstance>/<FieldsTopicName>`, `ara.com://services/<ServiceInterface>/<Major>.<Minor>/<FieldsTopicName>` where:
    * `ServiceInterface` takes the value of `serviceInterfaceId` (through `serviceInterfaceDeployment`)
    * `ServiceInstance` takes the value of `serviceInstanceId`
    * `Major` and `Minor` takes the value of `majorVersion` and `minorVersion` (via `serviceInterfaceDeployment`)
    * `MethodsTopicName` takes the value of `methodRequestTopicName` (through `serviceInterfaceDeployment`)
    * `FieldsTopicName` takes the value of `fieldRequestTopicName` (through `serviceInterfaceDeployment`)

⌋

[TR_DDSS_00204] Allow/publish rules for Required Service Instances

Status: DRAFT

⌈For each `DdsServiceInstanceToMachineMapping` referencing a `DdsSecureComProps` object, each associated `DdsRequiredServiceInstance` shall extend the associated (in the context of [TR_DDSS_00201]) Permissions Document `publish` element as follows:

- Under the `partitions` element:
  - Add an additional `partition` element with value `ara.com://services/<ServiceInterface>/<ServiceInstance>`, where:
    * `ServiceInterface` takes the value of `serviceInterfaceId` (through `serviceInterfaceDeployment`)
    * `ServiceInstance` takes the value of `requiredServiceInstanceId`
- Under the `topics` element:

  - Add four `topic` elements with values `ara.com://services/<ServiceInterface>/<ServiceInstance>/<MethodsTopicName>`, `ara.com://services/<ServiceInterface>/<Major>.<Minor>/<MethodsTopicName>`, `ara.com://services/<ServiceInterface>/<ServiceInstance>/<FieldsTopicName>`, `ara.com://services/<ServiceInterface>/<Major>.<Minor>/<FieldsTopicName>` where:
    * `ServiceInterface` takes the value of `serviceInterfaceId` (through `serviceInterfaceDeployment`)
    * `ServiceInstance` takes the value of `requiredServiceInstanceId`
    * `Major` and `Minor` takes the value of `majorVersion` and `minorVersion` (via `serviceInterfaceDeployment`)
    * `MethodsTopicName` takes the value of `methodRequestTopicName` (through `serviceInterfaceDeployment`)
    * `FieldsTopicName` takes the value of `fieldRequestTopicName` (through `serviceInterfaceDeployment`)

⌋

[TR_DDSS_00205] Allow/subscribe rules for Required Service Instances

Status: DRAFT

⌈For each `DdsServiceInstanceToMachineMapping` referencing a `DdsSecureComProps` object, each associated `DdsRequiredServiceInstance` shall extend the associated (in the context of [TR_DDSS_00201]) Permissions Document `subscribe` element as follows:

- Under the `partitions` element:
  - Add, if it doesn't exist yet, an empty `partition` element (for monitoring the discovery topic)
  - Add an additional `partition` element with value `ara.com://services/<ServiceInterface>/<ServiceInstance>`, where:
    * `ServiceInterface` takes the value of `serviceInterfaceId` (through `serviceInterfaceDeployment`)
    * `ServiceInstance` takes the value of `requiredServiceInstanceId`
- Under the `topics` element:
  - Add, if it doesn't exist yet, a `topic` element with value `ara.com://services/discovery` (for monitoring the discovery topic)

- Add two `topic` elements for each `ComEventGrant` referencing the current `DdsRequiredServiceInstance` via `serviceInstance` with values `ara.com://services/<ServiceInterface>/<ServiceInstance>/<EventTopicName>` and `ara.com://services/<ServiceInterface>/<Major>.<Minor>/<EventTopicName>` where:
  - `ServiceInterface` takes the value of `serviceInterfaceId` (through `serviceInterfaceDeployment`)
  - `ServiceInstance` takes the value of `requiredServiceInstanceId`
  - `Major` and `Minor` takes the value of `majorVersion` and `minorVersion` (via `serviceInterfaceDeployment`)
  - `EventTopicName` takes the value of `topicName` (through `serviceDeployment`)
- Add two `topic` elements, similar to the aforementioned `ComEventGrant` elements, for each `ComFieldGrant` referencing a field with `hasNotifier` set to `True` via `serviceDeployment`
- Add four `topic` elements with values `ara.com://services/<ServiceInterface>/<ServiceInstance>/<MethodsTopicName>`, `ara.com://services/<ServiceInterface>/<Major>.<Minor>/<MethodsTopicName>`, `ara.com://services/<ServiceInterface>/<ServiceInstance>/<FieldsTopicName>`, `ara.com://services/<ServiceInterface>/<Major>.<Minor>/<FieldsTopicName>` where:
  - `ServiceInterface` takes the value of `serviceInterfaceId` (through `serviceInterfaceDeployment`)
  - `ServiceInstance` takes the value of `requiredServiceInstanceId`
  - `Major` and `Minor` takes the value of `majorVersion` and `minorVersion` (via `serviceInterfaceDeployment`)
  - `MethodsTopicName` takes the value of `methodReplyTopicName` (through `serviceInterfaceDeployment`)
  - `FieldsTopicName` takes the value of `fieldReplyTopicName` (through `serviceInterfaceDeployment`)

⌋

A Mentioned Class Tables

For the sake of completeness, this chapter contains a set of class tables representing meta-classes mentioned in the context of this document.

Class	***AdaptivePlatformServiceInstance*** (abstract)			
Note	This meta-class represents the ability to describe the existence and configuration of a service instance in an abstract way. This Class is only used by the AUTOSAR Adaptive Platform.			
Base	*ARElement*, *ARObject*, *CollectableElement*, *Identifiable*, *MultilanguageReferrable*, *Packageable Element*, *Referrable*, *UploadableDesignElement*, *UploadablePackageElement*			
Subclasses	*ProvidedApServiceInstance*, *RequiredApServiceInstance*			
Aggregated by	ARPackage.element			
Attribute	***Type***	***Mult.***	***Kind***	***Note***
e2eEvent ProtectionProps	End2EndEvent ProtectionProps	*	aggr	This aggregation allows to protect an event or a field notifier that is defined inside of the ServiceInterface that is referenced by the ServiceInstance in the role service Interface.
e2eMethod ProtectionProps	End2EndMethod ProtectionProps	*	aggr	This aggregation allows to protect a method or a field getter or a field setter that is defined inside of the Service Interface that is referenced by the ServiceInstance in the role serviceInterface
secureCom Config	ServiceInterface ElementSecureCom Config	*	aggr	Configuration settings to secure the communication of ServiceInterface elements.
serviceInterface Deployment	ServiceInterface Deployment	0..1	ref	Reference to a ServiceInterfaceDeployment that identifies the ServiceInterface that is represented by the Service Instance.

Table A.1: AdaptivePlatformServiceInstance

Class	**ComEventGrant**			
Note	This meta-class represents the ability to grant access to a ServiceInterface.event. **Tags:** atp.Status=candidate atp.recommendedPackage=Grants This Class is only used by the AUTOSAR Adaptive Platform.			
Base	*ARElement*, *ARObject*, *CollectableElement*, *ComGrant*, *Grant*, *Identifiable*, *MultilanguageReferrable*, *PackageableElement*, *Referrable*, *UploadableDeploymentElement*, *UploadablePackageElement*			
Aggregated by	ARPackage.element			
Attribute	***Type***	***Mult.***	***Kind***	***Note***
design	ComEventGrantDesign	0..1	ref	This reference identifies the ComEventGrantDesign that the enclosing ComEventGrant was created from. **Stereotypes:** atpUriDef **Tags:** atp.Status=candidate
service Deployment	ServiceEvent Deployment	0..1	ref	This reference identifies the applicable deployment within the context of an AdaptivePlatformServiceInstance for which the grant applies. **Tags:** atp.Status=candidate

Table A.2: ComEventGrant

Class	**ComFieldGrant**			
Note	This meta-class represents the ability to grant access to a ServiceInterface.field. **Tags:** atp.Status=candidate atp.recommendedPackage=Grants This Class is only used by the AUTOSAR Adaptive Platform.			
Base	*ARElement*, *ARObject*, *CollectableElement*, *ComGrant*, *Grant*, *Identifiable*, *MultilanguageReferrable*, *PackageableElement*, *Referrable*, *UploadableDeploymentElement*, *UploadablePackageElement*			
Aggregated by	ARPackage.element			
Attribute	***Type***	***Mult.***	***Kind***	***Note***
design	ComFieldGrantDesign	0..1	ref	This reference identifies the ComFieldGrantDesign that the enclosing ComFieldGrant was created from. **Stereotypes:** atpUriDef **Tags:** atp.Status=candidate
role	FieldAccessEnum	0..1	attr	This attribute provides the ability to further specify the access to the ServiceInterface.field. **Tags:** atp.Status=candidate
service Deployment	ServiceField Deployment	0..1	ref	This reference identifies the applicable deployment within the context of an AdaptivePlatformServiceInstance for which the grant applies. **Tags:** atp.Status=candidate

Table A.3: ComFieldGrant

Class	***ComGrant*** (abstract)			
Note	This meta-class serves as the abstract base class for defining specific ComGrants **Tags:** atp.Status=candidate This Class is only used by the AUTOSAR Adaptive Platform.			
Base	*ARElement*, *ARObject*, *CollectableElement*, *Grant*, *Identifiable*, *MultilanguageReferrable*, *Packageable Element*, *Referrable*, *UploadableDeploymentElement*, *UploadablePackageElement*			
Subclasses	ComEventGrant, ComFieldGrant, ComMethodGrant, ComTriggerGrant			
Aggregated by	ARPackage.element			
Attribute	***Type***	***Mult.***	***Kind***	***Note***
remoteSubject	AbstractIamRemote Subject	*	ref	This optional reference defines the remoteSubject that is allowed to access the defined Object via the Grant. **Tags:** atp.Status=candidate
serviceInstance	AdaptivePlatform ServiceInstance	0..1	ref	This reference identifies the applicable AdaptivePlatform ServiceInstance for which the grant applies. **Tags:** atp.Status=candidate

Table A.4: ComGrant

Class	**CryptoCertificate**			
Note	This meta-class represents the ability to model a cryptographic certificate. This Class is only used by the AUTOSAR Adaptive Platform.			
Base	*ARObject*, *Identifiable*, *MultilanguageReferrable*, *Referrable*			
Aggregated by	CryptoModuleInstantiation.cryptoCertificate			
Attribute	***Type***	***Mult.***	***Kind***	***Note***
isPrivate	Boolean	0..1	attr	This attribute controls the possibility to access the content of the CryptoCertificateSlot by Find() interfaces of the X509 Provider.

Table A.5: CryptoCertificate

Class	**CryptoKeySlot**			
Note	This meta-class represents the ability to define a concrete key to be used for a crypto operation. **Tags:** atp.ManifestKind=MachineManifest This Class is only used by the AUTOSAR Adaptive Platform.			
Base	*ARObject*, *Identifiable*, *MultilanguageReferrable*, *Referrable*			
Aggregated by	CryptoProvider.keySlot			
Attribute	***Type***	***Mult.***	***Kind***	***Note***
algorithm Description	CryptoAlgorithm Description	*	aggr	This aggregation contains the collection of crypto algorithm descriptions that can be used in the context of the enclosing crypto key slot. **Tags:** atp.Status=candidate
allocateShadow Copy	Boolean	0..1	attr	This attribute defines whether a shadow copy of this Key Slot shall be allocated to enable rollback of a failed Key Slot update campaign (see interface BeginTransaction).
cryptoAlgId	String	0..1	attr	This attribute defines a crypto algorithm restriction (kAlgId Any means without restriction). The algorithm can be specified partially: family & length, mode, padding. Future Crypto Providers can support some crypto algorithms that are not well known/ standardized today, therefore AUTOSAR doesn't provide a concrete list of crypto algorithms' identifiers and doesn't suppose usage of numerical identifiers. Instead of this a provider supplier should provide string names of supported algorithms in accompanying documentation. The name of a crypto algorithm shall follow the rules defined in the specification of cryptography for Adaptive Platform.
cryptoKeySlot Design	CryptoKeySlotDesign	0..1	ref	This reference identifies the CryptoKeySlotDesign from which the referencing CryptoKeySlot was derived.
cryptoObject Type	CryptoObjectTypeEnum	0..1	attr	Object type that can be stored in the slot. If this field contains "Undefined" then mSlotCapacity must be provided and larger then 0. **Tags:** atp.Status=candidate
keySlotAllowed Modification	CryptoKeySlotAllowed Modification	0..1	aggr	Restricts how this keySlot may be used **Tags:** atp.Status=candidate
keySlotContent AllowedUsage	CryptoKeySlotContent AllowedUsage	*	aggr	Restriction of allowed usage of a key stored to the slot. **Tags:** atp.Status=candidate
slotCapacity	PositiveInteger	0..1	attr	Capacity of the slot in bytes to be reserved by the stack vendor. One use case is to define this value in case that the cryptoObjectType is undefined and the slot size can not be deduced from cryptoObjectType and cryptoAlgId. "0" means slot size can be deduced from cryptoObject Type and cryptoAlgId.
slotType	CryptoKeySlotType Enum	0..1	attr	This attribute defines whether the keySlot is exclusively used by the Application; or whether it is used by Stack Services and managed by a Key Manager Application. **Tags:** atp.Status=candidate

Table A.6: CryptoKeySlot

Class	**DdsEventDeployment**			
Note	DDS configuration settings for an Event. This Class is only used by the AUTOSAR Adaptive Platform.			
Base	*ARObject*, *Identifiable*, *MultilanguageReferrable*, *Referrable*, *ServiceEventDeployment*, *ServiceInterface DeploymentElement*			
Aggregated by	DdsFieldDeployment.notifier, *ServiceInterfaceDeployment*.eventDeployment			
Attribute	***Type***	***Mult.***	***Kind***	***Note***
eventTopic AccessRule	DdsTopicAccessRule	0..1	ref	DDS Security access rule applicable to the DDS Topics used for the service interface event.

▽

△

Class	**DdsEventDeployment**			
topicName	String	0..1	attr	Name of the DDS Topic associated with the Event.
transport Protocol	String	*	attr	This attribute defines over which Transport Layer Protocol(s) this event is intended to be sent.

Table A.7: DdsEventDeployment

Class	**DdsProvidedServiceInstance**			
Note	This meta-class represents the ability to describe the existence and configuration of a provided service instance in a concrete implementation on top of DDS. **Tags:** atp.recommendedPackage=ServiceInstances This Class is only used by the AUTOSAR Adaptive Platform.			
Base	*ARElement*, *ARObject*, *AdaptivePlatformServiceInstance*, *CollectableElement*, *DdsQosProps*, *DdsServiceInstanceProps*, *Identifiable*, *MultilanguageReferrable*, *PackageableElement*, *ProvidedApServiceInstance*, *Referrable*, *UploadableDesignElement*, *UploadablePackageElement*			
Aggregated by	ARPackage.element			
Attribute	***Type***	***Mult.***	***Kind***	***Note***
discoveryType	DdsServiceInstance DiscoveryTypeEnum	0..1	attr	Discovery protocol.
eventQosProps	DdsEventQosProps	*	aggr	List of configuration properties for the Events that are provided by the Service Instance.
fieldNotifierQos Props	DdsFieldQosProps	*	aggr	List of configuration properties for Field notifiers that are provided by the Service Instance.
resource IdentifierType	DdsServiceInstance ResourceIdentifierType Enum	0..1	attr	Type of resource identification scheme.
serviceInstance Id	PositiveInteger	0..1	attr	Identification number that is used by DDS to identify DomainParticipants associated with an instance of the service.

Table A.8: DdsProvidedServiceInstance

Class	***DdsQosProps*** (abstract)			
Note	QoS configuration properties for the DDS entities associated with an event, method, or field provided by or requested from a Service Instance using DDS as the underlying network binding. This Class is only used by the AUTOSAR Adaptive Platform.			
Base	*ARObject*			
Subclasses	DdsEventQosProps, DdsFieldQosProps, *DdsServiceInstanceProps*			
Attribute	***Type***	***Mult.***	***Kind***	***Note***
qosProfile	String	0..1	attr	Identifies a group of QoS Policies that apply to the DDS entities associated with the event, method, field, or the service instance.

Table A.9: DdsQosProps

Class	**DdsRequiredServiceInstance**
Note	This meta-class represents the ability to describe the existence and configuration of a required service instance in a concrete implementation on top of DDS. **Tags:** atp.recommendedPackage=ServiceInstances This Class is only used by the AUTOSAR Adaptive Platform.

▽

△

Class	**DdsRequiredServiceInstance**			
Base	*ARElement*, *ARObject*, *AdaptivePlatformServiceInstance*, *CollectableElement*, *DdsQosProps*, *DdsServiceInstanceProps*, *Identifiable*, *MultilanguageReferrable*, *PackageableElement*, *Referrable*, *RequiredApServiceInstance*, *UploadableDesignElement*, *UploadablePackageElement*			
Aggregated by	ARPackage.element			
Attribute	***Type***	***Mult.***	***Kind***	***Note***
blocklisted Version	DdsServiceVersion	*	aggr	Collection of blocklisted versions.
discoveryType	DdsServiceInstance DiscoveryTypeEnum	0..1	attr	Discovery protocol.
eventQosProps	DdsEventQosProps	*	aggr	List of configuration properties for the Events that are required by the Service Instance.
fieldNotifierQos Props	DdsFieldQosProps	*	aggr	List of configuration properties for Field notifiers that are required by the Service Instance.
requiredService InstanceId	AnyServiceInstanceId	0..1	attr	This attribute represents the ability to describe the required service instance ID.

Table A.10: DdsRequiredServiceInstance

Class	**DdsSecureComProps**			
Note	Identity and governance information of participants in case of DDS Security. **Tags:** atp.recommendedPackage=SecureComProps This Class is only used by the AUTOSAR Adaptive Platform.			
Base	*ARElement*, *ARObject*, *CollectableElement*, *Identifiable*, *MultilanguageReferrable*, *Packageable Element*, *Referrable*, *SecureComProps*, *UploadableDesignElement*, *UploadablePackageElement*			
Aggregated by	ARPackage.element			
Attribute	***Type***	***Mult.***	***Kind***	***Note***
governance	DdsSecureGovernance	0..1	ref	This attribute defines general DDS Security communication properties applicable to the DDS domain(s) in which the subject operates. **Tags:** atp.Status=candidate
identity	CryptoCertificate	0..1	ref	This attribute defines the cryptographic identity of the subject.

Table A.11: DdsSecureComProps

Class	**DdsSecureGovernance**			
Note	Configuration of DDS Security for all applications joining a specific set of DDS Domains. **Tags:** atp.Status=candidate atp.recommendedPackage=DdsSecureGovernances This Class is only used by the AUTOSAR Adaptive Platform.			
Base	*ARElement*, *ARObject*, *CollectableElement*, *Identifiable*, *MultilanguageReferrable*, *Packageable Element*, *Referrable*, *UploadableDesignElement*, *UploadablePackageElement*			
Aggregated by	ARPackage.element			
Attribute	***Type***	***Mult.***	***Kind***	***Note***
allowUnauthen-ticated Participants	Boolean	0..1	attr	Defines whether unauthenticated participants can join this domain. **Tags:** atp.Status=candidate
discovery ProtectionKind	DdsProtectionKind Enum	0..1	attr	Defines the kind of cryptographic transformation to apply in DDS discovery communication. **Tags:** atp.Status=candidate

▽

△

Class	**DdsSecureGovernance**			
domainId	DdsDomainRange	*	aggr	Set of domains to be covered by this property set. **Tags:** atp.Status=candidate
enableJoin AccessControl	Boolean	0..1	attr	Defines whether access control is to be enforced upon joining this domain. **Tags:** atp.Status=candidate
identity Certificate Authority	CryptoCertificate	0..1	ref	Certificate representing the identity certificate authority applicable to the domain(s) specified by domainsIds. **Tags:** atp.Status=candidate
liveliness ProtectionKind	DdsProtectionKind Enum	0..1	attr	Defines the kind of cryptographic transformation to apply in DDS liveliness communication. **Tags:** atp.Status=candidate
permission Certificate Authority	CryptoCertificate	0..1	ref	Certificate representing the permissions certificate authority applicable to the domain(s) specified by domainsIds. **Tags:** atp.Status=candidate
rtpsProtection Kind	DdsProtectionKind Enum	0..1	attr	Defines the kind of cryptographic transformation to apply to whole DDS RTPS. **Tags:** atp.Status=candidate

Table A.12: DdsSecureGovernance

Class	**DdsServiceInstanceToMachineMapping**			
Note	This meta-class allows to map DdsServiceInstances to a CommunicationConnector of a Machine. **Tags:** atp.recommendedPackage=ServiceInstanceToMachineMappings This Class is only used by the AUTOSAR Adaptive Platform.			
Base	*ARElement*, *ARObject*, *CollectableElement*, *Identifiable*, *MultilanguageReferrable*, *Packageable Element*, *Referrable*, *ServiceInstanceToMachineMapping*, *UploadableDesignElement*, *Uploadable PackageElement*			
Aggregated by	ARPackage.element			
Attribute	*Type*	*Mult.*	*Kind*	*Note*
secureCom PropsForDds	DdsSecureComProps	0..1	ref	Reference to SecureComProps applicable to the service instance.

Table A.13: DdsServiceInstanceToMachineMapping

Class	**DdsServiceInterfaceDeployment**			
Note	DDS configuration settings for a ServiceInterface. **Tags:** atp.recommendedPackage=ServiceInterfaceDeployments This Class is only used by the AUTOSAR Adaptive Platform.			
Base	*ARElement*, *ARObject*, *CollectableElement*, *Identifiable*, *MultilanguageReferrable*, *PackageableElement*, *Referrable*, *ServiceInterfaceDeployment*, *UploadableDesignElement*, *UploadablePackageElement*			
Aggregated by	ARPackage.element			
Attribute	*Type*	*Mult.*	*Kind*	*Note*
fieldReplyTopic Name	String	0..1	attr	Name of the DDS Reply Topic associated with the Field.
fieldRequest TopicName	String	0..1	attr	Name of the DDS Request Topic associated with the Field.
fieldTopics AccessRule	DdsTopicAccessRule	0..1	ref	DDS Security access rule applicable to the DDS Topics used for service interface field access methods (Get, Set).
methodReply TopicName	String	0..1	attr	Name of the DDS Reply Topic associated with the Method.

▽

△

Class	**DdsServiceInterfaceDeployment**			
methodRequest TopicName	String	0..1	attr	Name of the DDS Request Topic associated with the Method.
methodTopics AccessRule	DdsTopicAccessRule	0..1	ref	DDS Security access rule applicable to the DDS Topics used for service interface methods.
serviceInterface Id	String	0..1	attr	Unique Identifier that identifies the ServiceInterface in DDS. This Identifier is encoded in the USER_DATA QoS of the DomainParticipant associated with the Service Instance and its value is propagated by DDS Discovery messages.
transport Protocol	String	*	attr	This attribute defines over which Transport Layer Protocol(s) this Method is intended to be sent.

Table A.14: DdsServiceInterfaceDeployment

Class	**Field**			
Note	This meta-class represents the ability to define a piece of data that can be accessed with read and/or write semantics. It is also possible to generate a notification if the value of the data changes. This Class is only used by the AUTOSAR Adaptive Platform.			
Base	*ARObject*, *AtpFeature*, *AtpPrototype*, *AutosarDataPrototype*, *DataPrototype*, *Identifiable*, *Multilanguage Referrable*, *Referrable*			
Aggregated by	ApplicationInterface.attribute, *AtpClassifier*.atpFeature, ServiceInterface.field			
Attribute	***Type***	***Mult.***	***Kind***	***Note***
hasGetter	Boolean	0..1	attr	This attribute controls whether read access is foreseen to this field.
hasNotifier	Boolean	0..1	attr	This attribute controls whether a notification semantics is foreseen to this field.
hasSetter	Boolean	0..1	attr	This attribute controls whether write access is foreseen to this field.

Table A.15: Field

Class	**Process**			
Note	This meta-class provides information required to execute the referenced `Executable`. **Tags:** atp.recommendedPackage=Processes This Class is only used by the AUTOSAR Adaptive Platform.			
Base	*ARElement*, *ARObject*, *AbstractExecutionContext*, *AtpClassifier*, *CollectableElement*, *Identifiable*, *MultilanguageReferrable*, *PackageableElement*, *Referrable*, *UploadableDeploymentElement*, *Uploadable PackageElement*			
Aggregated by	ARPackage.element			
Attribute	***Type***	***Mult.***	***Kind***	***Note***
design	ProcessDesign	0..1	ref	This reference represents the identification of the design-time representation for the Process that owns the reference.
executable	Executable	*	ref	Reference to executable that is executed in the process. **Stereotypes:** atpUriDef
functionCluster Affiliation	String	0..1	attr	This attribute specifies which functional cluster the Process is affiliated with.
numberOf RestartAttempts	PositiveInteger	0..1	attr	This attribute defines how often a process shall be restarted if the start fails. numberOfRestartAttempts = "0" OR Attribute not existing, start once numberOfRestartAttempts = "1", start a second time

▽

△

Class	**Process**			
preMapping	Boolean	0..1	attr	This attribute describes whether the executable is preloaded into the memory.
processState Machine	ModeDeclarationGroup Prototype	0..1	aggr	Set of Process States that are defined for the process. This attribute is used to support the modeling of execution dependencies that utilize the condition of process state. Please note that the process states may not be modeled arbitrarily at any stage of the AUTOSAR workflow because the supported states are standardized in the context of the SWS Execution Management [4].
stateDependent StartupConfig	StateDependentStartup Config	*	aggr	Applicable startup configurations.

Table A.16: Process

Class	***ServiceFieldDeployment*** (abstract)			
Note	This abstract meta-class represents the ability to specify a deployment of a Field to a middleware transport layer. This Class is only used by the AUTOSAR Adaptive Platform.			
Base	*ARObject*, *Identifiable*, *MultilanguageReferrable*, *Referrable*, *ServiceInterfaceDeploymentElement*			
Subclasses	DdsFieldDeployment, SomeipFieldDeployment, UserDefinedFieldDeployment			
Aggregated by	*ServiceInterfaceDeployment*.fieldDeployment			
Attribute	***Type***	***Mult.***	***Kind***	***Note***
field	Field	0..1	ref	Reference to a Field that is deployed to a middleware transport layer. **Stereotypes:** atpUriDef

Table A.17: ServiceFieldDeployment

Class	**ServiceInterface**			
Note	This represents the ability to define a PortInterface that consists of a heterogeneous collection of methods, events and fields. **Tags:** atp.recommendedPackage=ServiceInterfaces This Class is only used by the AUTOSAR Adaptive Platform.			
Base	*ARElement*, *ARObject*, *AtpBlueprint*, *AtpBlueprintable*, *AtpClassifier*, *AtpType*, *CollectableElement*, *Identifiable*, *MultilanguageReferrable*, *PackageableElement*, *PortInterface*, *Referrable*			
Aggregated by	ARPackage.element			
Attribute	***Type***	***Mult.***	***Kind***	***Note***
event	VariableDataPrototype	*	aggr	This represents the collection of events defined in the context of a ServiceInterface. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=event.shortName, event.variationPoint.shortLabel vh.latestBindingTime=blueprintDerivationTime xml.sequenceOffset=30
field	Field	*	aggr	This represents the collection of fields defined in the context of a ServiceInterface. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=field.shortName, field.variationPoint.shortLabel vh.latestBindingTime=blueprintDerivationTime xml.sequenceOffset=40

▽

△

Class	**ServiceInterface**			
majorVersion	PositiveInteger	0..1	attr	Major version of the service contract. **Tags:** xml.sequenceOffset=10
method	ClientServerOperation	*	aggr	This represents the collection of methods defined in the context of a ServiceInterface. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=method.shortName, method.variation Point.shortLabel vh.latestBindingTime=blueprintDerivationTime xml.sequenceOffset=50
minorVersion	PositiveInteger	0..1	attr	Minor version of the service contract. **Tags:** xml.sequenceOffset=20
trigger	Trigger	*	aggr	This represents the collection of triggers defined in the context of a ServiceInterface. **Stereotypes:** atpSplitable; atpVariation **Tags:** atp.Splitkey=trigger.shortName, trigger.variation Point.shortLabel vh.latestBindingTime=blueprintDerivationTime xml.sequenceOffset=60

Table A.18: ServiceInterface

B Changed Specification Items

B.1 Release 25-11

B.1.1 Added Specification Items in R25-11

none

B.1.2 Changed Specification Items in R25-11

none

B.1.3 Deleted Specification Items in R25-11

none

B.2 Release 24-11

B.2.1 Added Specification Items in R24-11

none

B.2.2 Changed Specification Items in R24-11

none

B.2.3 Deleted Specification Items in R24-11

none