

Document Title	Collection of constraints on AUTOSAR M1 models
Document Owner	AUTOSAR
Document Responsibility	AUTOSAR
Document Identification No	635

Document Status	published
Part of AUTOSAR Standard	Foundation
Part of Standard Release	R21-11

Document Change History			
Date	Release	Changed by	Description
2021-11-25	R21-11	AUTOSAR Release Management	<ul style="list-style-type: none"> Updated constraints according to changes in TPS documents
2020-11-30	R20-11	AUTOSAR Release Management	<ul style="list-style-type: none"> Updated constraints according to changes in TPS documents Removed all SWS constraints Split document into 3 documents, reflecting the standards CP, AP, FO
2019-11-28	R19-11	AUTOSAR Release Management	<ul style="list-style-type: none"> Updated constraints according to changes in SWS and TPS documents Changed Document Status from Final to published
2018-10-31	4.4.0	AUTOSAR Release Management	Completion of constraint context by adding tables and classtables referenced by model constraints to this document
2017-12-08	4.3.1	AUTOSAR Release Management	minor corrections / clarifications / editorial changes; For details please refer to the ChangeDocumentation
2016-11-30	4.3.0	AUTOSAR Release Management	minor corrections / clarifications / editorial changes; For details please refer to the ChangeDocumentation
2015-07-31	4.2.2	AUTOSAR Release Management	minor corrections / clarifications / editorial changes; For details please refer to the ChangeDocumentation

2014-10-31	4.2.1	AUTOSAR Release Management	Editorial changes
2013-10-31	4.1.2	AUTOSAR Release Management	Updated constraints according to changes in SWS and TPS documents
2013-03-15	4.1.1	AUTOSAR Administration	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.

Table of Contents

1	Document Information and Content	6
2	Autosar Model Constraints	6
2.1	TPS_AbstractPlatformSpecification	6
2.2	TPS_FeatureModelExchangeFormat	8
2.3	TPS_LogAndTraceExtract	14
2.4	TPS_SecurityExtractTemplate	16
A	Mentioned Class Tables	18

References

1 Document Information and Content

This auxiliary document provides a collection of constraints for AUTOSAR models. All constraints are copied from template specification from the AUTOSAR Foundation, so this document does not introduce any new constraints.

A list of the documents that the constraints originate from can be found in the table of contents. Chapter 2 contains the collected constraints, grouped by source documents. All constraints from the same source document are contained within a single section.

2 Autosar Model Constraints

2.1 TPS_AbstractPlatformSpecification

[constr_6803]{DRAFT} Standardized values of [CompositionSwComponentType.category](#) [In a [System](#) with the [category](#) set to ABSTRACT_PLATFORM_SYSTEM_DESCRIPTION, any [CompositionSwComponentType](#) which is referenced by a [SwComponentPrototype](#) in the role [type](#) shall have the [category](#) set to:

- XP_COMPONENT_APPLICATION

]()

[constr_6806]{DRAFT} Standardized values of [ApplicationInterface.category](#) [The [category](#) of a [ApplicationInterface](#) can be set to either:

- XP_PORT_SECURITY
- XP_PORT_TIMESYNC
- XP_PORT_STORAGE
- XP_PORT_APPLICATION
- XP_PORT_SAFETY

]()

[constr_6807]{DRAFT} Exclusivity of a [ApplicationInterface](#) to an Abstract Platform [A [ApplicationInterface](#) shall not type a [PortPrototype](#) unless the [category](#) of the [System](#) is ABSTRACT_PLATFORM_SYSTEM_DESCRIPTION.

]()

[constr_6810]{DRAFT} Applicable categories for data types in an abstract platform [Table 2.1 defines the applicable data type [categorys](#) relating to applicable meta-model classes.

]()

Category	Applicable to ...						Description
	ApplicationDataType	ApplicationDeferredDataType	ApplicationArrayDataType	ApplicationRecordDataType	ApplicationPrimitiveDataType	ApplicationRecordElement	ApplicationArrayElement
VALUE				x	x	x	Contains a single value.
STRUCTURE			x		x	x	Holds one or several further elements which can have different AutosarDataTypes .
STRING				x	x	x	Contains a single value interpreted as a text string (note that it appears as a single value for the application domain).
ARRAY		x			x	x	A fixed-sized array of sub-elements of the same type.
BOOLEAN				x	x	x	Contains a single boolean (true/false) state.

Table 2.1: Usage of [category](#) for Data Types

[constr_6812]{DRAFT} [SwDataDefProps](#) applicable to [ApplicationDataTypes](#) exclusive to the abstract platform [A complete list of the allowed [SwDataDefProps](#) attributes and their multiplicities which are allowed for a given [category](#) is shown in table 2.2.

]()

Attributes of SwDataDefProps	Root Elem.				Attribute Existence per Category				
	ApplicationDataType	ApplicationDeferredDataType	ApplicationRecordElement	ApplicationArrayElement	VALUE	STRUCTURE	ARRAY	STRING	BOOLEAN
annotation	x	x	x	x	*	*	*	*	*
compuMethod	x				0..1				0..1
dataConstr.dataConstrRule.physConstrs	x		x	x	0..1		0..1		0..1
dataConstr.dataConstrRule.internalConstrs	x		x	x	d/c ¹		d/c		d/c
displayFormat	x		x	x	0..1		0..1	0..1	0..1
invalidValue	x				0..1			0..1	0..1
swTextProps	x							1	
unit	x				0..1			0..1	0..1
Other Attributes below the Root Element									
element: ApplicationRecordElement	x		x	x		1..*			
element: ApplicationArrayElement	x		x	x			1		
ApplicationArrayElement.arraySizeSemantics	x						0..1		
ApplicationArrayElement.maxNumberOfElements	x						1		

Table 2.2: Allowed Attributes vs. **category** for **ApplicationDataTypes**

[constr_6814]{DRAFT} **Restriction of ApplicationDeferredDataType.category** [The **category** of an **ApplicationDeferredDataType** shall be unassigned/undefined.

]()

2.2 TPS_FeatureModelExchangeFormat

[constr_3657] **Multiplicity of FMAttributeDef.max and FMAttributeDef.min** [For each **FMAttributeDef** the attributes **max** and **min** shall exist.

]()

[constr_3658] **Multiplicity of FMFeatureDecomposition.category** [For each **FMFeatureDecomposition** the attribute **category** shall exist.

]()

¹don't care

[constr_3659] Multiplicity of `FMFeatureDecomposition.feature` [For each `FMFeatureDecomposition` at least one reference in the role `feature` shall exist.

]()

[constr_3660] Multiplicity of `FMFeatureRelation.feature` [For each `FMFeatureRelation` at least one reference in the role `feature` shall exist.

]()

[constr_3661] Multiplicity of `FMFeatureSelection.feature` [For each `FMFeatureSelection` the reference in the role `feature` shall exist.

]()

[constr_3662] Multiplicity of `FMFeatureSelection.state` [For each `FMFeatureSelection` the attribute `state` shall exist.

]()

[constr_3663] Multiplicity of `FMAtributeValue.definition` [For each `FMAtributeValue` the reference in the role `definition` shall exist.

]()

[constr_3664] Multiplicity of `FMAtributeValue.value` [For each `FMAtributeValue` the attribute `value` shall exist.

]()

[constr_3665] Multiplicity of `FMFormulaByFeaturesAndAttributes.attribute` [For each `FMFormulaByFeaturesAndAttributes` the reference in the role `attribute` shall exist.

]()

[constr_3666] Multiplicity of `FMFormulaByFeaturesAndAttributes.feature` [For each `FMFormulaByFeaturesAndAttributes` the reference in the role `feature` shall exist.

]()

[constr_3667] Multiplicity of `FMFormulaByFeaturesAndSwSystemconsts.feature` [For each `FMFormulaByFeaturesAndSwSystemconsts` the reference in the role `feature` shall exist.

]()

[constr_5001] `FMFeatureRelation` shall not establish self-references [A `FMFeatureRelation` that is aggregated by a `FMFeature` f shall not reference f in the role `feature`. In other words: self-references are not allowed.

]()

[constr_5002] FMFeatureSelectionSet shall not have cycles in the include relation [Let S be a FMFeatureSelectionSet and let G be the inclusion graph for all FMFeatureSelectionSets as defined in [TPS_FMDT_00032]. There shall be no cycles in the inclusion graph.

]()

[constr_5003] FMFeatureSelectionSet shall not overwrite the state of included features [Let S be a FMFeatureSelectionSet that aggregates a FMFeatureSelection that has the state s and which refers to a FMFeature f in the role feature. Furthermore, let S_1 be a FMFeatureSelectionSet that aggregates a FMFeatureSelection that has the state s_1 and refers to the same FMFeature f in the role feature. Finally assume that S refers to S_1 in the role include.

Then the following conditions shall hold:

1. If the value of the attribute state of s_1 is undecided, then the value of the attribute state of s may be one of selected, deselected, and undecided.
2. If the value of the attribute state of s_1 is selected or deselected, then the value of the attribute state of s shall be the same as the attribute state in s_1 , or undecided.
3. Any other constellation is considered an error.

]()

[constr_5005] FMFeature shall not be referenced from more than one FMFeatureDecomposition [Let f be a FMFeature that is referenced from a FMFeatureDecomposition in the role feature. Then no other FMFeatureDecomposition shall reference f in the role feature.

]()

[constr_5007] FMFeature shall only be referenced from one FMFeatureModel in the role feature [Let f be a FMFeature, and F, F' be FMFeatureModels where F references f in the role feature, and F' also references f in the role feature. Then $F = F'$.

]()

[constr_5008] If present, the root feature shall be part of the feature model [Let r be the FMFeature referenced from FMFeatureModel in the role root, and $\{f_1, f_2, \dots, f_n\}$ the set of features referenced from the same FMFeatureModel in the role feature.

Then the following condition shall hold: $r \in \{f_1, f_2, \dots, f_n\}$.

]()

[constr_5009] Root feature shall be present if and only if the feature model is not empty [If a FMFeatureModel refers to one or more FMFeature elements in the role

`feature`, then exactly one of them shall be referenced by `FMFeatureModel` in the role `root`.

On the contrary, if `FMFeatureModel` does not refer to any `FMFeatures` in the role `feature`, then `root` shall be empty.

]()

[constr_5010] `FMFeatureDecomposition` may refer to a root feature of another feature model, but only once. [Let f_A be a `FMFeature` that is referenced by `FMFeatureModel` A in the role `feature`, but is also referenced from a `FMFeatureDecomposition` that is aggregated by a `FMFeature` f_B in the role `decomposition`.

Furthermore, let B be the `FMFeatureModel` that references f_B in the role `feature` with $A \neq B$. That is, f_A and f_B belong to different feature models.

Then *both* the following conditions shall hold:

1. f_A is referenced from A in the role `root`.
2. There is no other `FMFeatureDecomposition` (neither in B nor in any other `FMFeatureModel`) that references f_B in the role `feature`.

]()

[constr_5011] `FMFormulaByFeaturesAndAttributes` can refer to `FMFeatures` and `FMAttributeDefs`, but not to system constants [A formula of class `FMFormulaByFeaturesAndAttributes` is an expression that can use `FMFeatures` and `FMAttributeDefs`, but is not allowed to use `SwSystemconsts`.

]()

[constr_5013] Attributes `min` and `max` of `FMFeatureDecomposition` reserved for `category` `MULTIPLEFEATURE` [The optional attributes `min` and `max` of `FMFeatureDecomposition` are only allowed to be present if the `category` of the `FMFeatureDecomposition` is `MULTIPLEFEATURE`.

]()

[constr_5018] `FMFeatureSelectionSet` shall not include the same feature twice [Let $\{s_1, s_2, \dots, s_n\}$ be the set of `FMFeatureSelection` elements that are aggregated by a `FMFeatureSelectionSet` in the role `selection`. Furthermore, for each s_i , let f_i be the `FMFeature` that is referred to in the role `feature`. Then the following condition shall hold true:

$$\forall i, j \in \{1, 2, \dots, n\} : i \neq j \Rightarrow f_i \neq f_j$$

]()

[constr_5019] `FMFeatureModel` shall not contain the same `FMFeature` twice [Let F be a `FMFeatureModel`, and let f, f' be `FMFeatures` that are referenced from F in the role `feature`. Then $f \neq f'$.

]()

[constr_5020] Every `FMFeature` shall be contained in a `FMFeatureModel` [For every `FMFeature` f , there shall be a `FMFeatureModel` that refers to f in the role `feature`.

]()

[constr_5021] The underlying graph of a feature model shall be a tree. [Let F be a `FMFeatureModel` and G be the underlying graph of F as defined in [TPS_FMDT_00034]. Then G shall be a tree. Hence, we also refer to G as the *underlying tree* of F .

]()

[constr_5022] The root feature of a `FMFeatureModel` refers to the root of the underlying tree. [Let F be a `FMFeatureModel` and G be the underlying tree of F as defined in [TPS_FMDT_00034]. Furthermore, let r be the `FMFeature` referred to by the `root` feature of the `FMFeatureModel`.

Then the node in G which corresponds to r is the root of the tree G .

]()

[constr_5023] `FMFeatureSelectionSet` may only refer to `FMFeatures` from the associated `FMFeatureModel` [Let S be a `FMFeatureSelectionSet`, and $\{f_1, f_2, \dots, f_n\}$ be its *feature set* ([TPS_FMDT_00009]). Furthermore, let $\{g_1, g_2, \dots, g_m\}$ be the combined *feature sets* of the `FMFeatureModels` to which S refers to in the role `featureModel`.

Then the following condition shall hold: $\{f_1, f_2, \dots, f_n\} \subseteq \{g_1, g_2, \dots, g_m\}$.

]()

[constr_5024] `FMFeatureSelectionSet` shall not include itself [Let S be a `FMFeatureSelectionSet` and let S' be the `FMFeatureSelectionSet` to which S refers to in the role `include`.

Then the following condition shall hold: $S \neq S'$.

]()

[constr_5025] `FMFeatureSelectionSet` shall not overwrite the state of included features [Let S be a `FMFeatureSelectionSet` that aggregates a `FMFeatureSelection` that has the `state` s and which refers to a `FMFeature` f in the role `feature`. Furthermore, let S_1 (S_2) be a `FMFeatureSelectionSet` that aggregates a `FMFeatureSelection` that has the `state` s_1 (s_2) and refers to the same `FMFeature` f in the role `feature`. Finally assume that S refers to S_1 and S_2 in the role `include`.

Then the following conditions shall hold:

1. If the values of the attributes `state` of s_1 and s_2 are both `undecided`, then the value of the attribute `state` of s may be `selected`, `deselected` or `undecided`.

2. If the value of the attribute `state` of s_1 is `undecided` and the value of the attribute `state` of s_2 is `selected` or `deselected`, then the value of the attribute `state` of s shall be the same as the attribute `state` in s_2 , or `undecided`.
3. If the value of the attribute `state` of s_2 is `undecided` and the value of the attribute `state` of s_1 is `selected` or `deselected`, then the value of the attribute `state` of s shall be the same as the attribute `state` in s_1 , or `undecided`.
4. If the values of the attributes `state` of s_1 and s_2 are both either `selected` or `deselected`, then the value of the attribute `state` of s shall be the same as in attribute s_1 , or `undecided`.
5. Any other constellation is considered an error.

⌋()

[constr_5026] Semantics of attributes `max` and `min` in class `FMAAttributeDef`

⌈The following conditions shall hold for all instances of the class `FMAAttributeDef`:

- $\min \leq \text{defaultValue} \leq \max$ (`min` and `max` are both closed intervals)
- $\min < \text{defaultValue} \leq \max$ (`min` is an open interval, `max` is a closed interval)
- $\min < \text{defaultValue} < \max$ (`min` and `max` are both open intervals)
- $\min \leq \text{defaultValue} < \max$ (`min` is a closed interval, `max` is an open interval)

⌋()

[constr_5027] Semantics of attributes `max` and `min` of `FMAAttributeDef` in class `FMAAttributeValue`

⌈Let v be the attribute `value` of an `FMAAttributeValue` V that refers to `FMAAttributeDef` D in the role `definition`. Furthermore, let \min and \max be the values of the attributes `min` and `max` of D .

The following condition shall hold true:

$$\min \leq v \leq \max$$

⌋()

[constr_5028] Only one `FMAAttributeValue` per `FMAAttributeDef`

⌈Let S be a `FMFeatureSelectionSet` whose `FMFeatureSelections` aggregate `FMAAttributeValues` $\{v_1, v_2, \dots, v_n\}$ in the role `attributeValue`. For each v_i , let f_i be the `FMFeature` to which v_i refers to in the role `attributeDef`. Then the following condition shall hold:

$$\forall i \in \{1, \dots, n\} : i \neq j \Rightarrow f_i \neq f_j$$

⌋()

2.3 TPS_LogAndTraceExtract

[constr_5098] Usage of `DltArgument.networkRepresentation` [Usage of `DltArgument.networkRepresentation` shall follow the restrictions given in table 2.3.

]()

Attributes of <code>SwDataDefProps</code>	<code>networkRepresentation</code>
<code>annotation</code>	N/A
<code>baseType</code>	D
<code>compuMethod</code>	D
<code>dataConstr</code>	D
<code>displayFormat</code>	D
<code>displayPresentation</code>	N/A
<code>invalidValue</code>	N/A
<code>swComparisonVariable</code>	N/A
<code>swHostVariable</code>	N/A
<code>swTextProps</code>	N/A
<code>unit</code>	D

Table 2.3: Allowed usage of attributes in case of a `dltArgumentEntry`

[constr_5294] Existence of `DltEcu.ecuId` [For each `DltEcu`, the attribute `ecuId` shall exist when the Log And Trace Extract is created.

]()

[constr_5295] Existence of `DltApplication.context` [Each `DltApplication` shall reference at least one `DltContext` in the role `context` when the Log And Trace Extract is created.

]()

[constr_5296] Existence of `DltApplication.applicationId` [For each `DltApplication`, the attribute `applicationId` shall exist when the Log And Trace Extract is created.

]()

[constr_5297] Existence of `DltApplication.applicationDescription` [For each `DltApplication`, the attribute `applicationDescription` shall exist when the Log And Trace Extract is created.

]()

[constr_5298] Existence of `DltContext.contextId` [For each `DltContext`, the attribute `contextId` shall exist when the Log And Trace Extract is created.

]()

[constr_5299] Existence of `DltContext.contextDescription` [For each `DltContext`, the attribute `contextDescription` shall exist when the Log And Trace Extract is created.

]()

[constr_5300] Existence of `DltContext.dltMessage` [Each `DltContext` shall reference at least one `DltMessage` in the role `dltMessage` when the Log And Trace Extract is created.

]()

[constr_5301] Existence of `DltMessage.messageId` [For each `DltMessage`, the attribute `messageId` shall exist when the Log And Trace Extract is created.

]()

[constr_5302] Restriction in usage of `DltArgument.optional` attribute [The `optional` attribute shall not be set in a `DltArgument` that represents an array dimension according to 2.4.

]()

Type	length	dltArgumentEntry	SwBaseType of top level Dlt Argument
Predefined Text	NA	NA	NA
primitive Type	NA	NA	D
String	D	NA	D
1-dimensional Array	D	NA	D
n-dimensional Array	D	D	D
Struct	NA	D	NA

Table 2.4: Allowed usage of attributes for description of payload data types

[constr_5303] Restriction of `baseTypeSize` of a `DltArgument` [The `baseTypeSize` in the `networkRepresentation` of a `DltArgument` is restricted to 8, 16, 32, and 64 Bits.

]()

[constr_5304] Datatype of an Array [The `dltArgumentEntry` that is aggregated by a `DltArgument` that has the `length` attribute set to a value (represents an Array) shall not define a `SwBaseType` in the `networkRepresentation` since the data type of the Array is described by the `SwBaseType` in the `networkRepresentation` of the aggregating `DltArgument`.

]()

[constr_5305] `CompuMethod` in `DltArgument.networkRepresentation` [The `CompuMethod` that is used in the `networkRepresentation` of a `DltArgument` is limited to `category` TEXTTABLE.

]()

2.4 TPS_SecurityExtractTemplate

[constr_5600]{DRAFT} Valid interval for attribute `SecurityEventDefinition.id` [The valid interval for attribute `SecurityEventDefinition.id` is 0..65535.

]()

[constr_5601]{DRAFT} Uniqueness of `SecurityEventDefinition.id` [Within the scope of an IDS, i.e. for all `SecurityEventDefinitions` referenced by the same `IdsDesign`, there shall be no attribute `id` of any other `SecurityEventDefinition` that has the same value.

]()

[constr_5602]{DRAFT} Valid interval for attribute `SecurityEventOneEveryNFilter.n` [The valid interval for attribute `SecurityEventOneEveryNFilter.n` is 1..65535.

]()

[constr_5603]{DRAFT} Valid interval for attribute `SecurityEventAggregationFilter.minimumIntervalLength` [The valid interval for attribute `SecurityEventAggregationFilter.minimumIntervalLength` is]0..INF[seconds.

]()

[constr_5604]{DRAFT} Valid interval for attribute `SecurityEventThresholdFilter.intervalLength` [The valid interval for attribute `SecurityEventThresholdFilter.intervalLength` is]0..INF[seconds.

]()

[constr_5605]{DRAFT} Valid interval for attribute `SecurityEventThresholdFilter.thresholdNumber` [The valid interval for attribute `SecurityEventThresholdFilter.thresholdNumber` is 1..INF[.

]()

[constr_5606]{DRAFT} Valid interval for attribute `IdsmRateLimitation.timeInterval` [The valid interval for attribute `IdsmRateLimitation.timeInterval` is 0..65535 seconds.

]()

[constr_5607]{DRAFT} Valid interval for attribute `IdsmRateLimitation.maxEventsInInterval` [The valid interval for attribute `IdsmRateLimitation.maxEventsInInterval` is 0.. $(2^{64} - 1)$.

]()

[constr_5608]{DRAFT} Valid interval for attribute `IdsmTrafficLimitation.timeInterval` [The valid interval for attribute `IdsmTrafficLimitation.timeInterval` is 0..65535 seconds.

]()

[constr_5609]{DRAFT} Valid interval for attribute [IdsmTrafficLimitation.maxBytesInInterval](#) [The valid interval for attribute [IdsmTrafficLimitation.maxBytesInInterval](#) is $0..(2^{64} - 1)$.

]()

[constr_5610]{DRAFT} Unambiguous definition of execution platform for an [IdsmInstance](#) [For the meta-class [IdsmInstance](#), either the reference in the role [ecuInstance](#) or the reference in the role [idsmModuleInstantiation](#) shall be defined in order to ensure that the platform (Classic or Adaptive) on which an [IdsmInstance](#) is targeted to run is unambiguously defined.

]()

[constr_5611]{DRAFT} Unambiguous configuration of platform-dependent signature support for an [IdsmInstance](#) [For the meta-class [IdsmInstance](#), either the aggregation of [IdsmSignatureSupportCp](#) or of [IdsmSignatureSupportAp](#) shall be defined in order to ensure that the platform-dependent signature support is unambiguously configured.

]()

[constr_5612]{DRAFT} Unambiguous definition of platform-dependent network configuration for an [IdsmInstance](#) [For the meta-class [IdsmInstance](#), either the configuration of one [GeneralPurposeIPdu](#) with `category="IDS"` (for the Classic Platform as specified in [TPS_SECXT_01038]) or the network configuration through the reference [idsmModuleInstantiation](#) (for the Adaptive Platform as specified in [TPS_SECXT_01039]) shall be defined in order to ensure that the platform-dependent network configuration is unambiguously defined.

]()

[constr_5613]{DRAFT} Unambiguous definition of [SecurityEventStateFilter](#) for CP or AP [For [SecurityEventStateFilter](#), either the references in the role [blockIfStateActiveCp](#) or the references in the role [blockIfStateActiveAp](#) shall be defined in order to ensure the unambiguous applicability of the [SecurityEventStateFilter](#) towards the Classic or the Adaptive Platform.

]()

[constr_5614]{DRAFT} Upper bound for multiplicity of [BlockStates](#) aggregated by [IdsmInstance](#) [For the meta-class [IdsmInstance](#), the maximum number of aggregated [BlockStates](#) in the role [blockState](#) shall be 16.

]()

[constr_5615]{DRAFT} Restriction of [SecurityEventStateFilter](#) referencing [BlockStates](#) on CP [For a [SecurityEventStateFilter](#) on the Classic Platform, the references in the role [blockIfStateActiveCp](#) shall only reference those [BlockStates](#) that are aggregated in the role [blockState](#) by the [IdsmInstance](#)

which is mapped (by [SecurityEventContextMapping](#)) to that [SecurityEvent-FilterChain](#) of which the [SecurityEventStateFilter](#) is part of.

]()

A Mentioned Class Tables

Class	ApplicationDataType (abstract)			
Package	M2::AUTOSARTemplates::SWComponentTemplate::Datatype::Datatypes			
Note	<p>ApplicationDataType defines a data type from the application point of view. Especially it should be used whenever something "physical" is at stake.</p> <p>An ApplicationDataType represents a set of values as seen in the application model, such as measurement units. It does not consider implementation details such as bit-size, endianness, etc.</p> <p>It should be possible to model the application level aspects of a VFB system by using ApplicationDataTypes only.</p>			
Base	ARElement, ARObject, AtpBlueprint, AtpBlueprintable, AtpClassifier, AtpType, AutosarDataType, CollectableElement, Identifiable , MultilanguageReferrable, PackageableElement, Referrable			
Subclasses	ApplicationCompositeDataType, ApplicationDeferredDataType , ApplicationPrimitiveDataType			
Attribute	Type	Mult.	Kind	Note
–	–	–	–	–

Table A.1: ApplicationDataType

Class	ApplicationDeferredDataType			
Package	M2::AUTOSARTemplates::AbstractPlatform			
Note	<p>An placeholder data type in which the precise application data type is deferred to a later stage.</p> <p>Tags: atp.Status=draft atp.recommendedPackage=ApplicationDataTypes</p>			
Base	ARElement, ARObject, ApplicationDataType , AtpBlueprint, AtpBlueprintable, AtpClassifier, AtpType, AutosarDataType, CollectableElement, Identifiable , MultilanguageReferrable, PackageableElement, Referrable			
Attribute	Type	Mult.	Kind	Note
–	–	–	–	–

Table A.2: ApplicationDeferredDataType

Class	ApplicationInterface			
Package	M2::AUTOSARTemplates::AbstractPlatform			
Note	<p>This represents the ability to define a PortInterface that consists of a composition of commands (method calls), indications (events) and attributes (fields)</p> <p>Tags: atp.Status=draft atp.recommendedPackage=Interfaces</p>			
Base	ARElement, ARObject, AtpBlueprint, AtpBlueprintable, AtpClassifier, AtpType, CollectableElement, Identifiable , MultilanguageReferrable, PackageableElement, PortInterface, Referrable			





Class	ApplicationInterface			
Attribute	Type	Mult.	Kind	Note
attribute	Field	*	aggr	This represents the set of attributes defined in the context of an Abstract Platform ApplicationInterface. Stereotypes: atpVariation Tags: atp.Status=draft vh.latestBindingTime=blueprintDerivationTime
command	ClientServerOperation	*	aggr	This represents the collection of commands or function calls (with optional data arguments) defined in the context of an ApplicationInterface. Stereotypes: atpVariation Tags: atp.Status=draft vh.latestBindingTime=blueprintDerivationTime
indication	VariableDataPrototype	*	aggr	This represents the collection of indication or events (with optional data argument) defined in the context of an ApplicationInterface. Stereotypes: atpVariation Tags: atp.Status=draft vh.latestBindingTime=blueprintDerivationTime

Table A.3: ApplicationInterface

Class	BaseTypeDirectDefinition			
Package	M2::MSR::AsamHdo::BaseTypes			
Note	This BaseType is defined directly (as opposite to a derived BaseType)			
Base	ARObject, BaseTypeDefinition			
Attribute	Type	Mult.	Kind	Note
baseTypeEncoding	BaseTypeEncodingString	0..1	attr	This specifies, how an object of the current BaseType is encoded, e.g. in an ECU within a message sequence. Tags: xml.sequenceOffset=90
baseTypeSize	PositiveInteger	0..1	attr	Describes the length of the data type specified in the container in bits. Tags: xml.sequenceOffset=70
byteOrder	ByteOrderEnum	0..1	attr	This attribute specifies the byte order of the base type. Tags: xml.sequenceOffset=110
memAlignment	PositiveInteger	0..1	attr	This attribute describes the alignment of the memory object in bits. E.g. "8" specifies, that the object in question is aligned to a byte while "32" specifies that it is aligned four byte. If the value is set to "0" the meaning shall be interpreted as "unspecified". Tags: xml.sequenceOffset=100
nativeDeclaration	NativeDeclarationString	0..1	attr	This attribute describes the declaration of such a base type in the native programming language, primarily in the Programming language C. This can then be used by a code generator to include the necessary declarations into a header file. For example BaseType with shortName: "MyUnsignedInt" native Declaration: "unsigned short"





Class	BaseTypeDirectDefinition			
				<p>Results in typedef unsigned short MyUnsignedInt;</p> <p>If the attribute is not defined the referring Implementation DataTypes will not be generated as a typedef by RTE.</p> <p>If a nativeDeclaration type is given it shall fulfill the characteristic given by basetypeEncoding and baseType Size.</p> <p>This is required to ensure the consistent handling and interpretation by software components, RTE, COM and MCM systems.</p> <p>Tags:xml.sequenceOffset=120</p>

Table A.4: BaseTypeDirectDefinition

Class	BlockState			
Package	M2::AUTOSARTemplates::SecurityExtractTemplate			
Note	<p>This meta-class defines a block state that is part of the collection of block states belonging to a specific IdsmInstance. The Idsm shall discard any reported security event that is mapped to a filter chain containing a SecurityEventStateFilter that references the block state which is currently active in the Idsm.</p> <p>Tags:atp.Status=draft</p>			
Base	ARObject, <i>Identifiable</i> , MultilanguageReferrable, Referrable			
Attribute	Type	Mult.	Kind	Note
–	–	–	–	–

Table A.5: BlockState

Class	CompositionSwComponentType			
Package	M2::AUTOSARTemplates::SWComponentTemplate::Composition			
Note	<p>A CompositionSwComponentType aggregates SwComponentPrototypes (that in turn are typed by Sw ComponentTypes) as well as SwConnectors for primarily connecting SwComponentPrototypes among each others and towards the surface of the CompositionSwComponentType. By this means, hierarchical structures of software-components can be created.</p> <p>Tags:atp.recommendedPackage=SwComponentTypes</p>			
Base	ARElement, ARObject, AtpBlueprint, AtpBlueprintable, AtpClassifier, AtpType, CollectableElement, <i>Identifiable</i> , MultilanguageReferrable, PackageableElement, Referrable, SwComponentType			
Attribute	Type	Mult.	Kind	Note
component	SwComponent Prototype	*	aggr	<p>Stereotypes: atpSplitable; atpVariation</p> <p>Tags: atp.Splitkey=component.shortName, component.variation Point.shortLabel vh.latestBindingTime=postBuild</p>





Class	CompositionSwComponentType			
connector	SwConnector	*	aggr	<p>SwConnectors have the principal ability to establish a connection among PortPrototypes. They can have many roles in the context of a CompositionSwComponentType. Details are refined by subclasses.</p> <p>The aggregation of SwConnectors is subject to variability with the purpose to support variant data flow.</p> <p>Stereotypes: atpSplittable; atpVariation</p> <p>Tags: atp.Splitkey=connector.shortName, connector.variation Point.shortLabel vh.latestBindingTime=postBuild</p>

Table A.6: CompositionSwComponentType

Class	CompuMethod			
Package	M2::MSR::AsamHdo::ComputationMethod			
Note	<p>This meta-class represents the ability to express the relationship between a physical value and the mathematical representation.</p> <p>Note that this is still independent of the technical implementation in data types. It only specifies the formula how the internal value corresponds to its physical pendant.</p> <p>Tags:atp.recommendedPackage=CompuMethods</p>			
Base	ARElement, ARObject, AtpBlueprint, AtpBlueprintable, CollectableElement, Identifiable, MultilanguageReferrable, PackageableElement, Referrable			
Attribute	Type	Mult.	Kind	Note
compuInternalToPhys	Compu	0..1	aggr	<p>This specifies the computation from internal values to physical values.</p> <p>Tags:xml.sequenceOffset=80</p>
compuPhysToInternal	Compu	0..1	aggr	<p>This represents the computation from physical values to the internal values.</p> <p>Tags:xml.sequenceOffset=90</p>
displayFormat	DisplayFormatString	0..1	attr	<p>This property specifies, how the physical value shall be displayed e.g. in documents or measurement and calibration tools.</p> <p>Tags:xml.sequenceOffset=20</p>
unit	Unit	0..1	ref	<p>This is the physical unit of the Physical values for which the CompuMethod applies.</p> <p>Tags:xml.sequenceOffset=30</p>

Table A.7: CompuMethod

Class	DltApplication			
Package	M2::AUTOSARTemplates::LogAndTraceExtract			
Note	This meta-class represents the application from which the log and trace message originates.			
Base	ARObject, Identifiable, MultilanguageReferrable, Referrable			
Attribute	Type	Mult.	Kind	Note
applicationDescription	String	0..1	attr	This attribute can be used to describe the applicationId that is used in the log and trace message in more detail.





Class	DltApplication			
applicationId	String	0..1	attr	This attribute identifies the SW-C/BSW module in the log and trace message.
context	DltContext	*	ref	Definition of ContextIds for the Application. Stereotypes: atpSplitable; atpVariation Tags: atp.Splitkey=context.dltContext, context.variation Point.shortLabel vh.latestBindingTime=systemDesignTime

Table A.8: DltApplication

Class	DltArgument			
Package	M2::AUTOSARTemplates::LogAndTraceExtract			
Note	This element defines an Argument in a DltMessage.			
Base	ARObject, Identifiable , MultilanguageReferrable, Referrable			
Attribute	Type	Mult.	Kind	Note
dltArgument Entry	DltArgument	*	aggr	This aggregation is used to describe subElements of a Dlt Argument that defines a Structure.
length	PositiveInteger	0..1	attr	Describes the DltArgument length in case of Arrays and Strings in number of BaseType.
network Representation	SwDataDefProps	0..1	aggr	Definition of the networkRepresentation of the Dlt Argument.
optional	Boolean	0..1	attr	This attribute defines whether the argument is optional or not. If set to True, the argument can be omitted from the payload of a DLT message.
predefinedText	Boolean	0..1	attr	This attribute defines whether the DltArgument is a predefinedText (Static Data).
variableLength	Boolean	0..1	attr	This attribute defines whether the length of the Dlt Argument is variable (determined at runtime) or not.

Table A.9: DltArgument

Class	DltContext			
Package	M2::AUTOSARTemplates::LogAndTraceExtract			
Note	This meta-class represents the Context that groups Log and Trace Messages that are generated by an application. Tags: atp.recommendedPackage=DltContexts			
Base	ARElement, ARObject, CollectableElement, Identifiable , MultilanguageReferrable, Packageable Element, Referrable			
Attribute	Type	Mult.	Kind	Note
context Description	String	0..1	attr	This attribute can be used to describe the contextId that is used in the log and trace message in more detail.
contextId	String	0..1	attr	This attribute is used to group log and trace messages produced by an application to distinguish functionality.





Class	DltContext			
dltMessage	DltMessage	*	ref	Group of Log and Trace Messages assigned to the Dlt Context Stereotypes: atpSplitable; atpVariation Tags: atp.Splitkey=dltMessage.dltMessage, dlt Message.variationPoint.shortLabel vh.latestBindingTime=systemDesignTime

Table A.10: DltContext

Class	DltEcu			
Package	M2::AUTOSARTemplates::LogAndTraceExtract			
Note	This element represents an Ecu or Machine that produces logging and tracing information. Tags: atp.recommendedPackage=DltEcus			
Base	ARElement, ARObject, CollectableElement, Identifiable , MultilanguageReferrable, PackageableElement, Referrable			
Attribute	Type	Mult.	Kind	Note
application	DltApplication	*	aggr	Application on DltEcu that provides log or trace data. Stereotypes: atpSplitable; atpVariation Tags: atp.Splitkey=application.shortName, application.variationPoint.shortLabel vh.latestBindingTime=systemDesignTime
eculd	String	0..1	attr	This attribute defines the name of the ECU for use within the Dlt protocol.

Table A.11: DltEcu

Class	DltMessage			
Package	M2::AUTOSARTemplates::LogAndTraceExtract			
Note	This element defines a DltMessage.			
Base	ARObject, Identifiable , MultilanguageReferrable, Referrable			
Attribute	Type	Mult.	Kind	Note
dltArgument (ordered)	DltArgument	*	aggr	Ordered collection of DltArguments in the DltMessage.
messageId	PositiveInteger	0..1	attr	This attribute defines the unique Id for the DltMessage.
messageLine Number	PositiveInteger	0..1	attr	This attribute describes the position in the source file in which this log message was called.
messageSource File	String	0..1	attr	This attribute describes the source file in which this log message was called.
messageType Info	String	0..1	attr	This attribute describes the message Type

Table A.12: DltMessage

Class	FMAAttributeDef			
Package	M2::AUTOSARTemplates::FeatureModelTemplate			
Note	This metaclass represents the ability to define attributes for a feature.			
Base	ARObject, Identifiable , MultilanguageReferrable , Referrable			
Attribute	Type	Mult.	Kind	Note
defaultValue	Numerical	0..1	attr	This represents the default value of the attribute.
max	Limit	0..1	attr	Maximum possible value for the value of this attribute
min	Limit	0..1	attr	Minimum possible value for the value of this attribute

Table A.13: FMAAttributeDef

Class	FMAAttributeValue			
Package	M2::AUTOSARTemplates::FeatureModelTemplate			
Note	This defines a value for the attribute that is referred to in the role definition.			
Base	ARObject			
Attribute	Type	Mult.	Kind	Note
definition	FMAAttributeDef	0..1	ref	This refers to the definition of this attribute. Stereotypes: atpIdentityContributor
value	Numerical	0..1	attr	This represents the value of this attribute.

Table A.14: FMAAttributeValue

Class	FMFeature			
Package	M2::AUTOSARTemplates::FeatureModelTemplate			
Note	A FMFeature describes an essential characteristic of a product. Each FMFeature is contained in exactly one FMFeatureModel. Tags: atp.recommendedPackage=FMFeatureModels			
Base	ARElement, ARObject, CollectableElement, Identifiable , MultilanguageReferrable , PackageableElement , Referrable			
Attribute	Type	Mult.	Kind	Note
attributeDef	FMAAttributeDef	*	aggr	This defines the attributes of the given feature.
decomposition	FMFeatureDecomposition	*	aggr	Lists the sub-features of a feature.
maximum IntendedBinding Time	BindingTimeEnum	0..1	attr	Defines an upper bound for the binding time of the variation points that are associated with the FMFeature. This attribute is meant as a hint for the development process.
minimum IntendedBinding Time	BindingTimeEnum	0..1	attr	Defines a lower bound for the binding time of the variation points that are associated with the FMFeature. This attribute is meant as a hint for the development process.
relation	FMFeatureRelation	*	aggr	Defines relations for FMFeatures, for example dependencies on other FMFeatures, or conflicts with other FMFeatures. A FMFeature can only be part of a FMFeatureSelectionSet if all its relations are fulfilled.





Class	FMFeature			
restriction	FMFeatureRestriction	*	aggr	Defines restrictions for FMFeatures. A FMFeature can only be part of a FMFeatureSelectionSet if at least one of its restrictions evaluates to true.

Table A.15: FMFeature

Class	FMFeatureDecomposition			
Package	M2::AUTOSARTemplates::FeatureModelTemplate			
Note	A FMFeatureDecomposition describes dependencies between a list of features and their parent feature (i.e., the FMFeature that aggregates the FMFeatureDecomposition). The kind of dependency is defined by the attribute category.			
Base	ARObject			
Attribute	Type	Mult.	Kind	Note
category	CategoryString	0..1	attr	The category of a FMFeatureDecomposition defines the type of dependency that is defined by the FMFeatureDecomposition. There are four different categories: MANDATORYFEATURE, OPTIONALFEATURE, ALTERNATIVEFEATURE, and MULTIPLEFEATURE.
feature	FMFeature	*	ref	The features that are affected by the dependency defined by the FMFeatureDecomposition.
max	PositiveInteger	0..1	attr	For a dependency of category MULTIPLEFEATURE, this defines the maximum number of features allowed.
min	PositiveInteger	0..1	attr	For a dependency of category MULTIPLEFEATURE, this defines the minimum number of features allowed.

Table A.16: FMFeatureDecomposition

Class	FMFeatureModel			
Package	M2::AUTOSARTemplates::FeatureModelTemplate			
Note	A Feature model describes the features of a product line and their dependencies. Feature models are an optional part of an AUTOSAR model. Tags: atp.recommendedPackage=FMFeatureModels			
Base	ARElement, ARObject, CollectableElement, Identifiable, MultilanguageReferrable, PackageableElement, Referrable			
Attribute	Type	Mult.	Kind	Note
feature	FMFeature	*	ref	"feature" holds the list of features of the feature model. No FMFeature may be contained twice in this list. Also, each FMFeature may be contained on only one feature model. Stereotypes: atp.Splitable Tags: atp.Splitkey=feature
root	FMFeature	0..1	ref	The features of a feature model define a tree. The attribute root points to the root of this tree.

Table A.17: FMFeatureModel

Class	FMFeatureRelation			
Package	M2::AUTOSARTemplates::FeatureModelTemplate			
Note	Defines relations for FMFeatures, for example dependencies on other FMFeatures, or conflicts with other FMFeatures. A FMFeature can only be part of a FMFeatureSelectionSet if all its relations are fulfilled.			
Base	ARObject, <i>Identifiable</i> , <i>MultilanguageReferrable</i> , <i>Referrable</i>			
Attribute	Type	Mult.	Kind	Note
feature	FMFeature	*	ref	The FMFeature that is targeted by this FMFeature Relation.
restriction	FMConditionByFeatures AndAttributes	0..1	aggr	If given, the condition shall evaluate to true, in order for the FMFeatureRelation to be active.

Table A.18: FMFeatureRelation

Class	FMFeatureSelection			
Package	M2::AUTOSARTemplates::FeatureModelTemplate			
Note	A FMFeatureSelection represents the state of a particular FMFeature within a FMFeatureSelectionSet.			
Base	ARObject, <i>Identifiable</i> , <i>MultilanguageReferrable</i> , <i>Referrable</i>			
Attribute	Type	Mult.	Kind	Note
attributeValue	FMAAttributeValue	*	aggr	This defines a value for the attribute that is referred to in the role definition. Note that a FMFeatureSelection cannot include two FMAAttributeValues that refer to the same FMAAttributeDef in the role definition. Tags: xml.sequenceOffset=50
feature	FMFeature	0..1	ref	The FMFeature whose state is defined by this FMFeature Selection. Tags: xml.sequenceOffset=10
maximum SelectedBinding Time	BindingTimeEnum	0..1	attr	Defines an upper bound for the binding time of the variation points that are associated with the FMFeature, and refines its maximumIntendedBindingTime. This attribute is meant as a hint for the development process. Tags: xml.sequenceOffset=40
minimum SelectedBinding Time	BindingTimeEnum	0..1	attr	Defines a lower bound for the binding time of the variation points that are associated with the FMFeature, and refines its minimumIntendedBindingTime. This attribute is meant as a hint for the development process. Tags: xml.sequenceOffset=30
state	FMFeatureSelection State	0..1	attr	Defines how the FMFeature that is described by this FMFeatureSelection contributes to the FMFeature SelectionSet. A FMFeature may have the state selected, deselected or undecided. Tags: xml.sequenceOffset=20

Table A.19: FMFeatureSelection

Class	FMFeatureSelectionSet			
Package	M2::AUTOSARTemplates::FeatureModelTemplate			
Note	A FMFeatureSelectionSet is a set of FMFeatures that describes a specific product. Tags: atp.recommendedPackage=FMFeatureModelSelectionSets			
Base	ARElement, ARObject, CollectableElement, Identifiable , MultilanguageReferrable, PackageableElement, Referrable			
Attribute	Type	Mult.	Kind	Note
featureModel	FMFeatureModel	*	ref	All FMFeatures in this FMFeatureSelectionSet shall be part of the referenced FMFeatureModel.
include	FMFeatureSelectionSet	*	ref	Each FMFeatureSelectionSet may include one or more FMFeatureSelectionSets. This establishes a hierarchy among FMFeatureSelectionSets. See constr_5003 and constr_5025 for details.
selection	FMFeatureSelection	*	aggr	The set of FMFeatureSelections of this FMFeatureSelectionSet.

Table A.20: FMFeatureSelectionSet

Enumeration	FMFeatureSelectionState
Package	M2::AUTOSARTemplates::FeatureModelTemplate
Note	Defines how a particular FMFeature contributes to a FMFSelectionSet.
Literal	Description
deselected	The feature is excluded from the selection. Tags: atp.EnumerationLiteralIndex=0
selected	The feature is included in the selection. Tags: atp.EnumerationLiteralIndex=1
undecided	It is not yet decided whether the feature shall be included into or excluded from the selection. Tags: atp.EnumerationLiteralIndex=2

Table A.21: FMFeatureSelectionState

Class	<<atpMixedString>> FMFormulaByFeaturesAndAttributes (abstract)			
Package	M2::AUTOSARTemplates::FeatureModelTemplate			
Note	An expression that has the syntax of the AUTOSAR formula language but uses only references to features or feature attributes (not system constants) as operands.			
Base	ARObject, FormulaExpression			
Subclasses	FMConditionByFeaturesAndAttributes			
Attribute	Type	Mult.	Kind	Note
attribute	FMAttributeDef	0..1	ref	An expression of type FMFormulaByFeaturesAndAttributes may refer to attributes of FMFeatures.
feature	FMFeature	0..1	ref	An expression of type FMFormulaByFeaturesAndAttributes may refer to FMFeatures.

Table A.22: FMFormulaByFeaturesAndAttributes

Class	<<atpMixedString>> FMFormulaByFeaturesAndSwSystemconstns (abstract)			
Package	M2::AUTOSARTemplates::FeatureModelTemplate			
Note	An expression that has the syntax of the AUTOSAR formula language and may use references to features or system constants as operands.			
Base	<i>ARObject, FormulaExpression, SwSystemconstDependentFormula</i>			
Subclasses	FMConditionByFeaturesAndSwSystemconstns			
Attribute	Type	Mult.	Kind	Note
feature	FMFeature	0..1	ref	An expression of type FMFormulaByFeaturesAndSwSystemconstns may refer to FMFeatures.

Table A.23: FMFormulaByFeaturesAndSwSystemconstns

Class	GeneralPurposeIPdu			
Package	M2::AUTOSARTemplates::SystemTemplate::Fibex::FibexCore::CoreCommunication			
Note	This element is used for AUTOSAR Pdus without attributes that are routed by the PduR. Please note that the category name of such Pdus is standardized in the AUTOSAR System Template. Tags: atp.recommendedPackage=Pdus			
Base	<i>ARObject, CollectableElement, FibexElement, IPdu, Identifiable, MultilanguageReferrable, PackageableElement, Pdu, Referrable</i>			
Attribute	Type	Mult.	Kind	Note
–	–	–	–	–

Table A.24: GeneralPurposeIPdu

Class	Identifiable (abstract)			
Package	M2::AUTOSARTemplates::GenericStructure::GeneralTemplateClasses::Identifiable			
Note	Instances of this class can be referred to by their identifier (within the namespace borders). In addition to this, Identifiables are objects which contribute significantly to the overall structure of an AUTOSAR description. In particular, Identifiables might contain Identifiables.			
Base	<i>ARObject, MultilanguageReferrable, Referrable</i>			
Subclasses	ARPackage, AbstractDolpLogicAddressProps, AbstractEvent, AbstractImplementationDataTypeElement, AbstractSecurityEventFilter, AbstractSecurityIdsmInstanceFilter, AbstractServiceInstance, AdaptiveModuleInstantiation, ApplicationEndpoint, ApplicationError, ArtifactChecksum, AtpBlueprint, AtpBlueprintable, AtpClassifier, AtpFeature, AutosarOperationArgumentInstance, AutosarVariableInstance, BlockState, BuildActionEntity, BuildActionEnvironment, Chapter, ClassContentConditional, ClientIdDefinition, ClientServerOperation, Code, CollectableElement, ComManagementMapping, CommConnectorPort, CommunicationConnector, CommunicationController, Compiler, ConsistencyNeeds, ConsumedEventGroup, CouplingPort, CouplingPortStructuralElement, CryptoKeySlot, CryptoServiceMapping, DataPrototypeGroup, DataTransformation, DependencyOnArtifact, DiagEventDebounceAlgorithm, DiagnosticConnectedIndicator, DiagnosticDataElement, DiagnosticDebounceAlgorithmProps, DiagnosticFunctionInhibitSource, DiagnosticRoutineSubfunction, DltApplication, DltArgument, DltMessage, DolpInterface, DolpLogicAddress, DolpRoutingActivation, EndToEndProtection, EthernetWakeupSleepOnDataLineConfig, EventHandler, ExclusiveArea, ExecutableEntity, ExecutionTime, FMAttributeDef, FMFeatureMapAssertion, FMFeatureMapCondition, FMFeatureMapElement, FMFeatureRelation, FMFeatureRestriction, FMFeatureSelection, FlexrayArTpNode, FlexrayTpPduPool, FrameTriggering, GeneralParameter, GlobalTimeGateway, GlobalTimeMaster, GlobalTimeSlave, HeapUsage, HwAttributeDef, HwAttributeLiteralDef, HwPin, HwPinGroup, IPsecRule, IPv6ExtHeaderFilterList, ISignalToIPduMapping, ISignalTriggering, IdentCaption, InternalTriggeringPoint, Keyword, LifecycleState, Linker, MacMulticastGroup, McDataInstance, MemorySection, ModeDeclaration, ModeDeclaration			





Class	Identifiable (abstract)			
	<p>Mapping, ModeSwitchPoint, NetworkEndpoint, <i>NmCluster</i>, <i>NmNode</i>, <i>PackageableElement</i>, Parameter Access, PduActivationRoutingGroup, PduToFrameMapping, PduTriggering, PerInstanceMemory, <i>PhysicalChannel</i>, PortGroup, <i>PortInterfaceMapping</i>, PossibleErrorReaction, ResourceConsumption, RootSwCompositionPrototype, RptComponent, RptContainer, RptExecutableEntity, RptExecutableEntity Event, RptExecutionContext, RptProfile, RptServicePoint, RunnableEntityGroup, <i>SdgAttribute</i>, <i>SdgClass</i>, SecureCommunicationAuthenticationProps, SecureCommunicationFreshnessProps, SecurityEvent ContextProps, <i>ServiceNeeds</i>, SignalServiceTranslationEventProps, SignalServiceTranslationProps, SocketAddress, SomeipTpChannel, <i>SpecElementReference</i>, <i>StackUsage</i>, StaticSocketConnection, StructuredReq, SwGenericAxisParamType, SwServiceArg, SwcServiceDependency, SystemMapping, <i>TimeBaseResource</i>, TimingCondition, <i>TimingConstraint</i>, <i>TimingDescription</i>, TimingExtensionResource, TimingModelInstance, Topic1, TpAddress, TraceableTable, TraceableText, <i>TracedFailure</i>, <i>Transformation Props</i>, TransformationTechnology, Trigger, VariableAccess, VariationPointProxy, ViewMap, VlanConfig, WaitPoint</p>			
Attribute	Type	Mult.	Kind	Note
adminData	AdminData	0..1	aggr	<p>This represents the administrative data for the identifiable object.</p> <p>Stereotypes: atpSplittable</p> <p>Tags: atp.Splitkey=adminData xml.sequenceOffset=-40</p>
annotation	Annotation	*	aggr	<p>Possibility to provide additional notes while defining a model element (e.g. the ECU Configuration Parameter Values). These are not intended as documentation but are mere design notes.</p> <p>Tags:xml.sequenceOffset=-25</p>
category	CategoryString	0..1	attr	<p>The category is a keyword that specializes the semantics of the Identifiable. It affects the expected existence of attributes and the applicability of constraints.</p> <p>Tags:xml.sequenceOffset=-50</p>
desc	MultiLanguageOverview Paragraph	0..1	aggr	<p>This represents a general but brief (one paragraph) description what the object in question is about. It is only one paragraph! Desc is intended to be collected into overview tables. This property helps a human reader to identify the object in question.</p> <p>More elaborate documentation, (in particular how the object is built or used) should go to "introduction".</p> <p>Tags:xml.sequenceOffset=-60</p>
introduction	DocumentationBlock	0..1	aggr	<p>This represents more information about how the object in question is built or is used. Therefore it is a DocumentationBlock.</p> <p>Tags:xml.sequenceOffset=-30</p>
uuid	String	0..1	attr	<p>The purpose of this attribute is to provide a globally unique identifier for an instance of a meta-class. The values of this attribute should be globally unique strings prefixed by the type of identifier. For example, to include a DCE UUID as defined by The Open Group, the UUID would be preceded by "DCE:". The values of this attribute may be used to support merging of different AUTOSAR models. The form of the UUID (Universally Unique Identifier) is taken from a standard defined by the Open Group (was Open Software Foundation). This standard is widely used, including by Microsoft for COM (GUIDs) and by many companies for DCE, which is based on CORBA. The method for generating these 128-bit IDs is published in the standard and the effectiveness and uniqueness of the IDs is not in practice disputed. If the id namespace is omitted, DCE is assumed. An example is</p>





Class	Identifiable (abstract)			
				<p>"DCE:2fac1234-31f8-11b4-a222-08002b34c003". The uuid attribute has no semantic meaning for an AUTOSAR model and there is no requirement for AUTOSAR tools to manage the timestamp.</p> <p>Tags:xml.attribute=true</p>

Table A.25: Identifiable

Class	IdsDesign			
Package	M2::AUTOSARTemplates::SecurityExtractTemplate			
Note	<p>This meta-class represents the root element of a SecurityExtract file for IDS development. It defines the scope of an IDS to be designed and implemented by referencing all SecurityExtract meta-classes that need to be included into the IDS development process.</p> <p>Tags: atp.Status=draft atp.recommendedPackage=IdsDesigns</p>			
Base	ARElement, ARObject, CollectableElement, Identifiable, MultilanguageReferrable, PackageableElement, Referrable			
Attribute	Type	Mult.	Kind	Note
element	IdsCommonElement	*	ref	<p>This reference includes an element with IDS related definitions into the IdsDesign.</p> <p>Stereotypes: atpSplitable; atpVariation</p> <p>Tags: atp.Splitkey=element.idsCommonElement, element.variationPoint.shortLabel atp.Status=draft vh.latestBindingTime=systemDesignTime</p>

Table A.26: IdsDesign

Class	IdsmInstance			
Package	M2::AUTOSARTemplates::SecurityExtractTemplate			
Note	<p>This meta-class provides the ability to create a relation between an EcuInstance and a specific class of filters for security events that apply for all security events reported on the referenced EcuInstance.</p> <p>Tags: atp.Status=draft atp.recommendedPackage=IdsmInstanceToEcuInstanceMappings</p>			
Base	ARElement, ARObject, CollectableElement, Identifiable, IdsCommonElement, MultilanguageReferrable, PackageableElement, Referrable			
Attribute	Type	Mult.	Kind	Note
blockState	BlockState	*	aggr	<p>This reference defines the BlockState in the collection BlockStateSet.</p> <p>Tags:atp.Status=draft</p>





Class	IdsmInstance			
eculInstance	EculInstance	0..1	ref	<p>This reference identifies the EculInstance whose security events (of any type) shall be limited by the specific class of filters.</p> <p>Stereotypes: atpVariation</p> <p>Tags: atp.Status=draft vh.latestBindingTime=systemDesignTime</p>
idsmInstancelId	PositiveInteger	0..1	attr	<p>This attribute is used to provide a source identification in the context of reporting security events..</p> <p>Tags:atp.Status=draft</p>
idsmModuleInstantiation	IdsmModuleInstantiation	0..1	ref	<p>This reference identifies the meta-class that defines the attributes for the IdsM configuration on a specific machine.</p> <p>Stereotypes: atpSplitable</p> <p>Tags: atp.Splitkey=idsmModuleInstantiation atp.Status=draft</p>
rateLimitationFilter	IdsmRateLimitation	0..1	ref	<p>This reference identifies the applicable rate limitation filter for all security events on the related EculInstance.</p> <p>Stereotypes: atpVariation</p> <p>Tags: atp.Status=draft vh.latestBindingTime=preCompileTime</p>
signatureSupportAp	IdsmSignatureSupportAp	0..1	aggr	<p>The existence of this aggregation specifies that the IdsM shall add a signature to the QSEv messages it sends onto the network. The cryptographic algorithm and key to be used for this signature is further specified by the aggregated meta-class specifically for the Adaptive Platform.</p> <p>Stereotypes: atpSplitable</p> <p>Tags: atp.Splitkey=signatureSupportAp atp.Status=draft</p>
signatureSupportCp	IdsmSignatureSupportCp	0..1	aggr	<p>The existence of this aggregation specifies that the IdsM shall add a signature to the QSEv messages it sends onto the network. The cryptographic algorithm and key to be used for this signature is further specified by the aggregated meta-class specifically for the Classic Platform.</p> <p>Stereotypes: atpSplitable</p> <p>Tags: atp.Splitkey=signatureSupportCp atp.Status=draft</p>
timestampFormat	String	0..1	attr	<p>The existence of this attribute specifies that the IdsM shall add a timestamp to the QSEv messages it sends onto the network. I.e., if this attribute does not exist, no timestamp shall be added to the QSEv messages.</p> <p>The content of this attribute further specifies the timestamp format as follows: - "AUTOSAR" defines AUTOSAR standardized timestamp format according to the Synchronized Time-Base Manager - Any other string defines a proprietary timestamp format.</p> <p>Note: A string defining a proprietary timestamp format shall be prefixed by a company-specific name fragment to avoid collisions.</p> <p>Tags:atp.Status=draft</p>





Class	IdsmInstance			
trafficLimitationFilter	IdsmTrafficLimitation	0..1	ref	<p>This reference identifies the applicable traffic limitation filter for all security events on the related EcuInstance.</p> <p>Stereotypes: atpVariation</p> <p>Tags: atp.Status=draft vh.latestBindingTime=preCompileTime</p>

Table A.27: IdsmInstance

Class	IdsmRateLimitation			
Package	M2::AUTOSARTemplates::SecurityExtractTemplate			
Note	<p>This meta-class represents the configuration of a rate limitation filter for security events. This means that security events are dropped if the number of events (of any type) processed within a configurable time window is greater than a configurable threshold.</p> <p>Tags:atp.Status=draft</p>			
Base	ARObject, AbstractSecurityIdsmInstanceFilter, Identifiable , MultilanguageReferrable , Referrable			
Attribute	Type	Mult.	Kind	Note
maxEventsInInterval	PositiveInteger	1	attr	<p>This attribute configures the threshold for dropping security events if the number of all processed security events exceeds the threshold in the respective time interval.</p> <p>Tags:atp.Status=draft</p>
timeInterval	Float	1	attr	<p>This attribute configures the length of the time interval in seconds for dropping security events if the number of all processed security events exceeds the configurable threshold within the respective time interval.</p> <p>Tags:atp.Status=draft</p>

Table A.28: IdsmRateLimitation

Class	IdsmSignatureSupportAp			
Package	M2::AUTOSARTemplates::SecurityExtractTemplate			
Note	<p>This meta-class defines, for the Adaptive Platform, the cryptographic algorithm and key to be used by the IdsM instance for providing signature information in QSEv messages.</p> <p>Tags:atp.Status=draft</p>			
Base	ARObject			
Attribute	Type	Mult.	Kind	Note
cryptoPrimitive	String	1	attr	<p>This attribute defines the cryptographic algorithm to be used for providing authentication information in QSEv messages. The content of this attribute shall comply to the "Cryptographic Primitives Naming Convention".</p> <p>Tags:atp.Status=draft</p>
keySlot	CryptoKeySlot	0..1	ref	<p>This reference denotes the cryptographic key to be used by the cryptographic algorithm for providing authentication information in QSEv messages.</p> <p>Tags:atp.Status=draft</p>

Table A.29: IdsmSignatureSupportAp

Class	IdsmSignatureSupportCp			
Package	M2::AUTOSARTemplates::SecurityExtractTemplate			
Note	<p>This meta-class defines, for the Classic Platform, the cryptographic algorithm and key to be used by the IdsM instance for providing signature information in QSEv messages.</p> <p>Tags:atp.Status=draft</p>			
Base	ARObject			
Attribute	Type	Mult.	Kind	Note
authentication	CryptoServicePrimitive	0..1	ref	<p>This reference denotes the cryptographic primitives for providing authentication information in QSEv messages.</p> <p>Tags:atp.Status=draft</p>
cryptoServiceKey	CryptoServiceKey	0..1	ref	<p>This reference denotes the cryptographic key to be used by the cryptographic algorithm for providing authentication information in QSEv messages.</p> <p>Tags:atp.Status=draft</p>

Table A.30: IdsmSignatureSupportCp

Class	IdsmTrafficLimitation			
Package	M2::AUTOSARTemplates::SecurityExtractTemplate			
Note	<p>This meta-class represents the configuration of a traffic limitation filter for Security Events. This means that security events are dropped if the size (in terms of bandwidth) of security events (of any type) processed within a configurable time window is greater than a configurable threshold.</p> <p>Tags:atp.Status=draft</p>			
Base	ARObject, AbstractSecurityIdsmInstanceFilter, <i>Identifiable</i> , MultilanguageReferrable, Referrable			
Attribute	Type	Mult.	Kind	Note
maxBytesInInterval	PositiveInteger	0..1	attr	<p>This attribute configures the threshold for dropping security events if the size of all processed security events exceeds the threshold in the respective time interval.</p> <p>Tags:atp.Status=draft</p>
timeInterval	Float	0..1	attr	<p>This attribute configures the length of the time interval in seconds for dropping security events if the size of all processed security events exceeds the configurable threshold within the respective time interval.</p> <p>Tags:atp.Status=draft</p>

Table A.31: IdsmTrafficLimitation

Class	PortPrototype (abstract)			
Package	M2::AUTOSARTemplates::SWComponentTemplate::Components			
Note	<p>Base class for the ports of an AUTOSAR software component.</p> <p>The aggregation of PortPrototypes is subject to variability with the purpose to support the conditional existence of ports.</p>			
Base	ARObject, AtpBlueprintable, AtpFeature, AtpPrototype, <i>Identifiable</i> , MultilanguageReferrable, Referrable			
Subclasses	AbstractProvidedPortPrototype, AbstractRequiredPortPrototype			
Attribute	Type	Mult.	Kind	Note





Class	PortPrototype (abstract)			
logAndTraceMessageCollectionSet	LogAndTraceMessageCollectionSet	0..1	ref	Reference to a collection of Log or Trace messages that will be used by the application. Tags: atp.Status=draft

Table A.32: PortPrototype

Class	SecurityEventAggregationFilter			
Package	M2::AUTOSARTemplates::SecurityExtractTemplate			
Note	This meta-class represents the aggregation filter that aggregates all security events occurring within a configured time frame into one (i.e. the last reported) security event. Tags: atp.Status=draft			
Base	ARObject, AbstractSecurityEventFilter, Identifiable , MultilanguageReferrable, Referrable			
Attribute	Type	Mult.	Kind	Note
contextDataSource	SecurityEventContextDataSourceEnum	0..1	attr	This attributes defines whether the context data of the first or last time-aggregated security event shall be used for the resulting qualified security event.
minimumIntervalLength	TimeValue	0..1	attr	This attribute represents the configuration of the minimum time window in seconds for the aggregation filter. Tags: atp.Status=draft

Table A.33: SecurityEventAggregationFilter

Class	SecurityEventContextMapping (abstract)			
Package	M2::AUTOSARTemplates::SecurityExtractTemplate			
Note	This meta-class represents the ability to create an association between a collection of security events, an IdsM instance which handles the security events and the filter chains applicable to the security events. Tags: atp.Status=draft			
Base	ARElement, ARObject, CollectableElement, Identifiable , IdsCommonElement, IdsMapping, MultilanguageReferrable, PackageableElement, Referrable			
Subclasses	SecurityEventContextMappingApplication, SecurityEventContextMappingBswModule, SecurityEventContextMappingCommConnector, SecurityEventContextMappingFunctionalCluster			
Attribute	Type	Mult.	Kind	Note
filterChain	SecurityEventFilterChain	0..1	ref	This reference defines the filter chain to be applied to each of the referenced security events (depending on the reporting mode). Stereotypes: atpVariation Tags: atp.Status=draft vh.latestBindingTime=preCompileTime
idsmInstance	IdsmInstance	0..1	ref	This reference defines the IdsmInstance onto which the security events are mapped. Stereotypes: atpVariation Tags: atp.Status=draft vh.latestBindingTime=systemDesignTime





Class	SecurityEventContextMapping (abstract)			
mappedSecurityEvent	SecurityEventContextProps	*	aggr	<p>This aggregation represents (through further references) the SecurityEventDefinitions to be mapped to an Idsm Instance with additional mapping-dependent properties.</p> <p>Stereotypes: atpSplitable; atpVariation</p> <p>Tags: atp.Splitkey=mappedSecurityEvent.shortName, mappedSecurityEvent.variationPoint.shortLabel atp.Status=draft vh.latestBindingTime=preCompileTime</p>

Table A.34: SecurityEventContextMapping

Class	SecurityEventDefinition			
Package	M2::AUTOSARTemplates::SecurityExtractTemplate			
Note	<p>This meta-class defines a security-related event as part of the intrusion detection system.</p> <p>Tags: atp.Status=draft atp.recommendedPackage=SecurityEventDefinitions</p>			
Base	ARElement, ARObject, CollectableElement, Identifiable, IdsCommonElement, MultilanguageReferrable, PackageableElement, Referrable			
Attribute	Type	Mult.	Kind	Note
eventSymbolName	SymbolProps	0..1	aggr	<p>This aggregation defines optionally an alternative Event Name for the SecurityEventDefinition in case there is a collision of shortNames.</p> <p>Stereotypes: atpSplitable</p> <p>Tags: atp.Splitkey=eventSymbolName.shortName atp.Status=draft</p>
id	PositiveInteger	0..1	attr	<p>This attribute represents the numerical identification of the defined security event. The identification shall be unique within the scope of the IDS.</p> <p>Tags:atp.Status=draft</p>

Table A.35: SecurityEventDefinition

Class	SecurityEventFilterChain			
Package	M2::AUTOSARTemplates::SecurityExtractTemplate			
Note	<p>This meta-class represents a configurable chain of filters used to qualify security events. The different filters of this filter chain are applied in the follow order: SecurityEventStateFilter, SecurityEventOneEveryNFilter, SecurityEventAggregationFilter, SecurityEventThresholdFilter.</p> <p>Tags: atp.Status=draft atp.recommendedPackage=SecurityFilterChains</p>			
Base	ARElement, ARObject, CollectableElement, Identifiable, IdsCommonElement, MultilanguageReferrable, PackageableElement, Referrable			
Attribute	Type	Mult.	Kind	Note
aggregation	SecurityEventAggregationFilter	0..1	aggr	<p>This aggregation represents the aggregation filter in the filter chain.</p> <p>Tags:atp.Status=draft</p>





Class	SecurityEventFilterChain			
oneEveryN	SecurityEventOneEveryNFilter	0..1	aggr	This aggregation represents the sampling filter in the filter chain. Tags: atp.Status=draft
state	SecurityEventStateFilter	0..1	aggr	This aggregation represents the state filter in the event chain. Tags: atp.Status=draft
threshold	SecurityEventThresholdFilter	0..1	aggr	This aggregation represents the threshold filter in the filter chain. Tags: atp.Status=draft

Table A.36: SecurityEventFilterChain

Class	SecurityEventOneEveryNFilter			
Package	M2::AUTOSARTemplates::SecurityExtractTemplate			
Note	This meta-class represents the configuration of a sampling (i.e. every n-th event is sampled) filter for security events. Tags: atp.Status=draft			
Base	ARObject, AbstractSecurityEventFilter, Identifiable , MultilanguageReferrable, Referrable			
Attribute	Type	Mult.	Kind	Note
n	PositiveInteger	0..1	attr	This attribute represents the configuration of the sampling filter, i.e. it configures the parameter "n" that controls how many events (n-1) shall be dropped after a sampled event until a new sample is created. Tags: atp.Status=draft

Table A.37: SecurityEventOneEveryNFilter

Class	SecurityEventStateFilter			
Package	M2::AUTOSARTemplates::SecurityExtractTemplate			
Note	This meta-class represents the configuration of a state filter for security events. The referenced states represent a block list, i.e. the security events are dropped if the referenced state is the active state in the relevant state machine (which depends on whether the IdsM instance runs on the Classic or the Adaptive Platform). Tags: atp.Status=draft			
Base	ARObject, AbstractSecurityEventFilter, Identifiable , MultilanguageReferrable, Referrable			
Attribute	Type	Mult.	Kind	Note
blockIfState ActiveAp	ModeDeclaration	*	iref	For the AP, this reference defines the machine states of the block list. That means, if a security event (mapped to the filter chain to which the SecurityEventStateFilter belongs to) is reported when the machine is in one of the block listed states, the IdsM shall discard the reported security event. Tags: atp.Status=draft InstanceRef implemented by: FunctionGroupStateIn FunctionGroupSetInstanceRef





Class	SecurityEventStateFilter			
blockIfState ActiveCp	BlockState	*	ref	For the CP, this reference defines the states of the block list. That means, if a security event (mapped to the filter chain to which the SecurityEventStateFilter belongs to) is reported when the currently active block state in the IdsM is one of the referenced block listed states, the IdsM shall discard the reported security event. Tags: atp.Status=draft

Table A.38: SecurityEventStateFilter

Class	SecurityEventThresholdFilter			
Package	M2::AUTOSARTemplates::SecurityExtractTemplate			
Note	This meta-class represents the threshold filter that drops (repeatedly at each beginning of a configurable time interval) a configurable number of security events . All subsequently arriving security events (within the configured time interval) pass the filter. Tags: atp.Status=draft			
Base	ARObject, AbstractSecurityEventFilter, Identifiable , MultilanguageReferrable, Referrable			
Attribute	Type	Mult.	Kind	Note
intervalLength	TimeValue	0..1	attr	This attribute configures the time interval in seconds for one threshold filter operation. Tags: atp.Status=draft
threshold Number	PositiveInteger	0..1	attr	This attribute configures the threshold number, i.e. how many security events in the configured time frame are dropped before subsequent events start to pass the filter. Tags: atp.Status=draft

Table A.39: SecurityEventThresholdFilter

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

Table A.40: SwBaseType

Class	SwComponentPrototype			
Package	M2::AUTOSARTemplates::SWComponentTemplate::Composition			
Note	Role of a software component within a composition.			
Base	ARObject, AtpFeature, AtpPrototype, Identifiable , MultilanguageReferrable, Referrable			
Attribute	Type	Mult.	Kind	Note





Class	SwComponentPrototype			
type	SwComponentType	0..1	tref	Type of the instance. Stereotypes: isOfType

Table A.41: SwComponentPrototype

Class	<<atpVariation>> SwDataDefProps			
Package	M2::MSR::DataDictionary::DataDefProperties			
Note	<p>This class is a collection of properties relevant for data objects under various aspects. One could consider this class as a "pattern of inheritance by aggregation". The properties can be applied to all objects of all classes in which SwDataDefProps is aggregated.</p> <p>Tags:vh.latestBindingTime=codeGenerationTime</p>			
Base	ARObject			
Attribute	Type	Mult.	Kind	Note
annotation	Annotation	*	aggr	<p>This aggregation allows to add annotations (yellow pads ...) related to the current data object.</p> <p>Tags: xml.roleElement=true xml.roleWrapperElement=true xml.sequenceOffset=20 xml.typeElement=false xml.typeWrapperElement=false</p>
baseType	SwBaseType	0..1	ref	<p>Base type associated with the containing data object.</p> <p>Tags:xml.sequenceOffset=50</p>
compuMethod	CompuMethod	0..1	ref	<p>Computation method associated with the semantics of this data object.</p> <p>Tags:xml.sequenceOffset=180</p>
dataConstr	DataConstr	0..1	ref	<p>Data constraint for this data object.</p> <p>Tags:xml.sequenceOffset=190</p>
displayFormat	DisplayFormatString	0..1	attr	<p>This property describes how a number is to be rendered e.g. in documents or in a measurement and calibration system.</p> <p>Tags:xml.sequenceOffset=210</p>
displayPresentation	DisplayPresentationEnum	0..1	attr	<p>This attribute controls the presentation of the related data for measurement and calibration tools.</p>
invalidValue	ValueSpecification	0..1	aggr	<p>Optional value to express invalidity of the actual data element.</p> <p>Tags:xml.sequenceOffset=255</p>
swComparisonVariable	SwVariableRefProxy	*	aggr	<p>Variables used for comparison in an MCD process.</p> <p>Tags: xml.sequenceOffset=170 xml.typeElement=false</p>
swHostVariable	SwVariableRefProxy	0..1	aggr	<p>Contains a reference to a variable which serves as a host-variable for a bit variable. Only applicable to bit objects.</p> <p>Tags: xml.sequenceOffset=220 xml.typeElement=false</p>
swTextProps	SwTextProps	0..1	aggr	<p>the specific properties if the data object is a text object.</p> <p>Tags:xml.sequenceOffset=120</p>





Class	<<atpVariation>> SwDataDefProps			
unit	Unit	0..1	ref	Physical unit associated with the semantics of this data object. This attribute applies if no compuMethod is specified. If both units (this as well as via compuMethod) are specified the units shall be compatible. Tags: xml.sequenceOffset=350

Table A.42: SwDataDefProps

Class	SwSystemconst			
Package	M2::MSR::DataDictionary::SystemConstant			
Note	This element defines a system constant which serves an input to select a particular variation point. In particular a system constant serves as an operand of the binding function (swSyscond) in a Variation point. Note that the binding process can only happen if a value was assigned to to the referenced system constants. Tags: atp.recommendedPackage=SwSystemconst			
Base	ARElement, ARObject, AtpDefinition, CollectableElement, <i>Identifiable</i> , MultilanguageReferrable, PackageableElement, Referrable			
Attribute	Type	Mult.	Kind	Note
swDataDef Props	SwDataDefProps	0..1	aggr	This denotes the data definition properties of the system constant. This supports to express the limits and optionally a conversion within the internal to physical values by a compu method. Tags: xml.sequenceOffset=40

Table A.43: SwSystemconst

Class	System			
Package	M2::AUTOSARTemplates::SystemTemplate			
Note	The top level element of the Abstract Platform System Description. Tags: atp.recommendedPackage=Systems			
Base	ARElement, ARObject, AtpClassifier, AtpFeature, AtpStructureElement, CollectableElement, <i>Identifiable</i> , MultilanguageReferrable, PackageableElement, Referrable			
Attribute	Type	Mult.	Kind	Note
mapping	SystemMapping	*	aggr	Stereotypes: atpSplitable; atpVariation Tags: atp.Splitkey=mapping.shortName, mapping.variationPoint.shortLabel vh.latestBindingTime=postBuild
rootSoftware Composition	RootSwComposition Prototype	0..1	aggr	Aggregation of the root software composition, containing all software components in the System in a hierarchical structure. Stereotypes: atpSplitable; atpVariation Tags: atp.Splitkey=rootSoftwareComposition.shortName, rootSoftwareComposition.variationPoint.shortLabel vh.latestBindingTime=systemDesignTime
systemVersion	RevisionLabelString	1	attr	Version number of the System Description.

Table A.44: System