IDTA 02011-1 Hierarchical Structures enabling Bills of Material – Extension based on IEC 81346

Version 1.0
August 2025

SPECIFICATION

Submodel Template of the Asset Administration Shell

Imprint

Publisher
Industrial Digital Twin Association
Lyoner Straße 18
60528 Frankfurt am Main
Germany
https://www.industrialdigitaltwin.org/

Version history

Date	Version	Comment
04.08.2025	1.0	Release of the official Submodel template published by IDTA.

Contents

Figures

Tables

1 General

1.1 About this document

This document is a part of a specification series. Each part specifies the contents of a Submodel Template for the Asset Administration Shell (AAS). The AAS is described in [1], [2], [3] and [6]. First exemplary Submodel contents were described in [4], while the actual format of this document was derived by the "Administration Shell in Practice" [5]. The format aims to be very concise, giving only minimal necessary information for applying a Submodel Template, while leaving deeper descriptions and specification of concepts, structures and mapping to the respective documents [1] to [6].

The target group of the specification are developers and editors of technical documentation and manufacturer information, which are describing assets in smart manufacturing by means of the Asset Administration Shell (AAS) and therefore need to create a Submodel instance with a hierarchy of SubmodelElements. This document especially details on the question, which SubmodelElements with which semantic identification shall be used for this purpose.

This Submodel is based on IEC 81346 Part 1 2021, further details can be found in the IEC; this version is also based on the Hierarchical Structures enabling Bills of Material Submodel in version 1.1. The numbering scheme used follows the structure: original Submodel number (IDTA 02011) - extension code (1). Hence the number IDTA 02011-1 for this Submodel extension.

Scope of the Submodel

The structuring of machines, plants or systems is an essential task in engineering to make the complexity of a technical system manageable. IEC 81346 standardises a concept of structuring based on various aspects that an object (asset) can have. In this sense, a system always consists of several objects, which means that the system (composite asset) also has corresponding aspects. Figure 1 shows three of the aspects as examples. The standard describes a total of five aspects:

- What an object is supposed to do or what it actually does – Function aspect (=)
- The means by which an object does what it is supposed to do – Product aspect (-)
- Planned or actual location of the object – Location aspect (+)
- Group of objects that have the same properties – Type aspect (%)
- A freely definable aspect that must be specified by the creator – Other aspect (#)

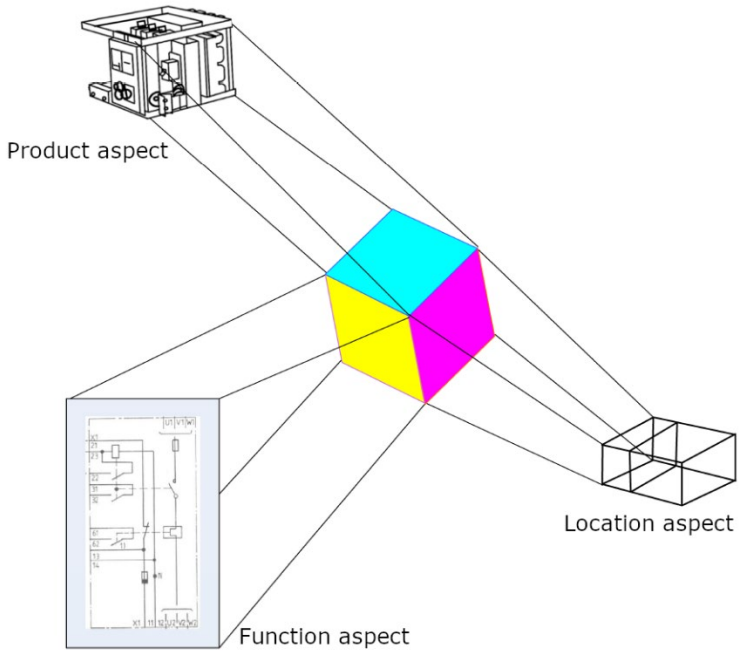

Figure 1 Different aspect views of a system, based on IEC 81346 [8]

Hierarchies are created by structuring the various aspects, which is why the Hierarchical Structures enabling Bills of Material Submodel is extended in accordance with the IEC standard. Each aspect can in turn have its own hierarchy.

This Submodel is based on IEC 81346 Part 1 2021, further details can be found in the IEC; this version is also based on the Hierarchical Structures enabling Bills of Material Submodel in version 1.1.

1.2 Relevant standards for the Submodel Template

1.2.1 Version of the Asset Administration Shell

This Submodel Template for the Asset Administration Shell is specified as a Submodel according to the AAS Part 1 – Metamodel **Version 3.0** [6]. It is designed to enable the interaction with distributed AAS as specified by the AAS Part 2 - Application Programming Interfaces **Version 3.0** [7].

1.3 Use cases, requirements and design decisions

1.3.1 Use Cases

Table 1: Use cases

Use Case	Explanation
Electrical Engineering	In the context of electrical design, structuring is an essential task in order to master the complexity of the machine to be designed. The resulting equipment identifier uniquely identifies each object (asset) within the system, which in turn is essential in downstream phases such as maintenance.
Fluidic Engineering	As with electrical design, structuring according to IEC 81346 is also used in fluidics.
Mechanical Engineering	The concepts of structuring can also be applied in the context of mechanical engineering, creating a relationship between the different engineering disciplines (electrical engineering, fluidics).

1.3.2 Design Decisions

As an extension of the Hierarchical Structures enabling Bills of Material Submodel, the same design decisions apply as for Hierarchical Structure enabling Bills of Material.

Various other design decisions were made during development, which are explained below.

A Property *PrefixLevel* is defined in the original *EntryNode* to differentiate between the various aspects (=, +, -, etc.). Mapping this property via the SupplementalSemID or the qualifier was not considered appropriate or too restrictive due to the open depth of possible aspects (=, ==, ===, ====, ...). From a current perspective, a maximum depth of three will be used in real scenarios, but this flexibility should be preserved for future requirements and thus the standard should not be restricted. It is therefore possible to define any depth via the *PrefixLevel*. The effect of the decision can be seen in section 2.4 in the *IEC81346EntryNode* class.

In addition, an optional *HierarchyPrefix* is introduced, which can be used for the evaluation of a reference designator in software tools. The IEC standard has two variants of aspect hierarchies in representation reference designator. One is with the *PrefixLevel* itself or a dot between the levels. Figure illustrates the situation using the functional aspect. This is to prevent the different software tools from always displaying the indicator in different ways. The definition can correspond to internal company specifications so that the same understanding prevails within a company.

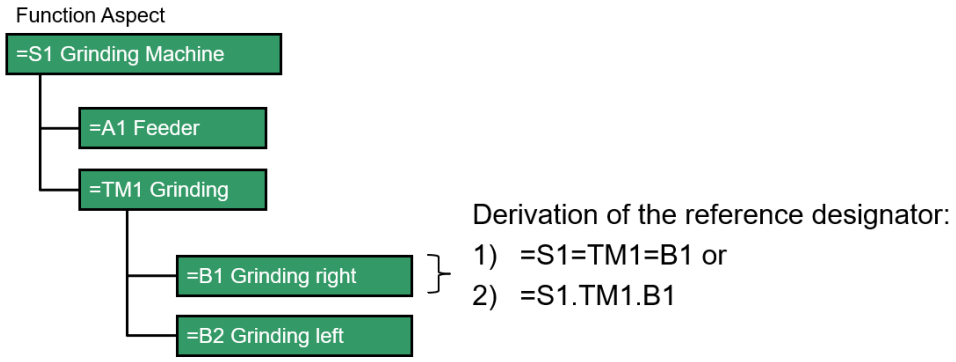

Figure 2 Derivation of the reference designator from the hierarchy levels and its representation.

IEC 81346 provides three levels for describing an aspect. Each level is described with a capital letters. The exact definition of the letters is described in Part 2 of the standard, see also Figure 3. A separate property was defined for each level for the detailed semantic description. This is based on the ECLASS classification, which also provides a separate property for each level. In addition to the levels, the defined counter must also be merged for the transfer to an equipment identifier. The following method applies, as described in IEC 81346:

a) A letter code followed by a number; or
b) A letter code; or
c) A number

The three levels and a property for the number are described further in section 2.5

2 Submodel Hierarchical Structures enabling Bills of Material – Extension based on IEC 81346

2.1 Approach

In this document a Submodel extension is defined. The Submodel is usable in asset administration shells for type and instance assets.

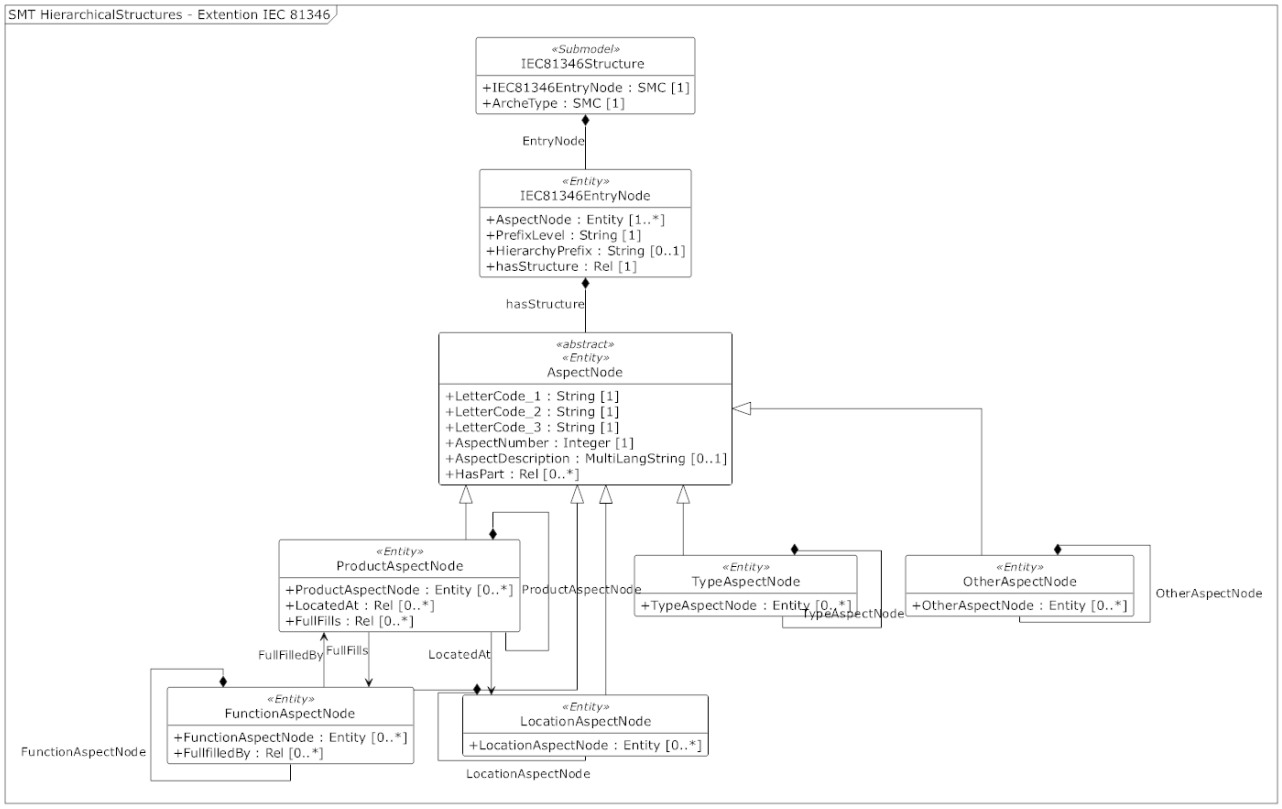

Figure 3 Submodel extension based on IEC 81346

2.2 Recommendation of Usage

2.2.1 Sematical extension of Nodes

Only one aspect can be described per Submodel characteristic. A *LocationAspectNode* can only contain *LocationAspectNodes*. The prefix (=, +, etc) determines the *AspectNode* types to be used. It is necessary to uniquely identify the described aspect. Therefore, the correct SemanticId must be used based on the aspect to be described. The SemanticId to be used are specified below:

Table 2 SemanticId of the various aspects

Aspect	IRI	Prefix
FunctionAspect	[IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/FunctionAspectNode/1/0	=
LocationAspect	[IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/LocationAspectNode/1/0	+

ProductAspect	[IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/ProductAspectNode/1/0	-
TypeAspect	[IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/TypeAspectNode/1/0	&
OtherAspect	[IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/OtherAspectNode/1/0	#

2.2.2 ArcheType handling

Nested relationships, as intended with the *ArcheType* property, can also be implemented with this Submodel extension. The hierarchy of structuring is distributed across the various assets, which themselves structure the assets they contain. A derivation of an asset identifier should follow defined rules that the hierarchy is attached to a specific aspect. These rules are not part of this Submodel.

2.2.3 External Usage

To reference this Submodel, various RelationElements are defined that should be used by another Submodel (e.g. Hierarchical Structures enabling Bills of Material). This means that is possible to derive a reference designator, as objects (assets) can refer to their associated structural aspects. For this purpose, a SubmodelCollection can summarise all references to this Submodel. Figure 4 shows an abstract example of how three aspects (Function, Location, Product) of this Submodel extension can be referenced by another (Hierachical Structure Submodel) and its entities. The required references are defined in the following table.

Table 3: Additional attribute of the Submodel

idShort:	ReferenceDesignation		
Class:	SubmodelElementCollection		
semanticId:	[IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/ReferenceDesignation/1/0		
Parent:	Entity		
Explanation:	Grouping of all reference designation references for simple structuring. This allows a reference designation to be assigned to an entity, which contains all the necessary references to this Submodel.		
Inherits from:	-		
[SME type]	**semanticId = [idType]value**	**[valueType]**	**card.**
idShort	**Description@en**	**example**	
[Rel] FunctionAspectRelationship__00__	[IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/FunctionAspectRelationship/1/0 Relationship to a function aspect. The first attribute contains the next higher entity, the second must be an entity with the SemanticId FunctionAspect.		0..*
[Rel] ProductAspectRelationship__00__	[IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/ProductAspectRelationship/1/0 Relationship to a product aspect.		0..*

	The first attribute contains the next higher entity, the second must be an entity with the SemanticId ProductAspect.		
[Rel] LocationAspectRelationship__00__	[IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/LocationAspectRelationship/1/0 Relationship to a location aspect The first attribute contains the next higher entity, the second must be an entity with the SemanticId LocationAspect.		0..*
[Rel] TypeAspectRelationship__00__	[IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/TypeAspectRelationship/1/0 Relationship to a type aspect The first attribute contains the next higher entity, the second must be an entity with the SemanticId TypeAspect.		0..*
[Rel] OtherAspectRelationship__00__	[IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/OtherAspectRelationship/1/0 Relationship to an other aspect The first attribute contains the next higher entity, the second must be an entity with the SemanticId OtherAspect.		0..*

All AspectReferences are non-hierarchical relationships and correspond to the semantics of a NonHierarchicalReferences from the OPC UA Specification [9].

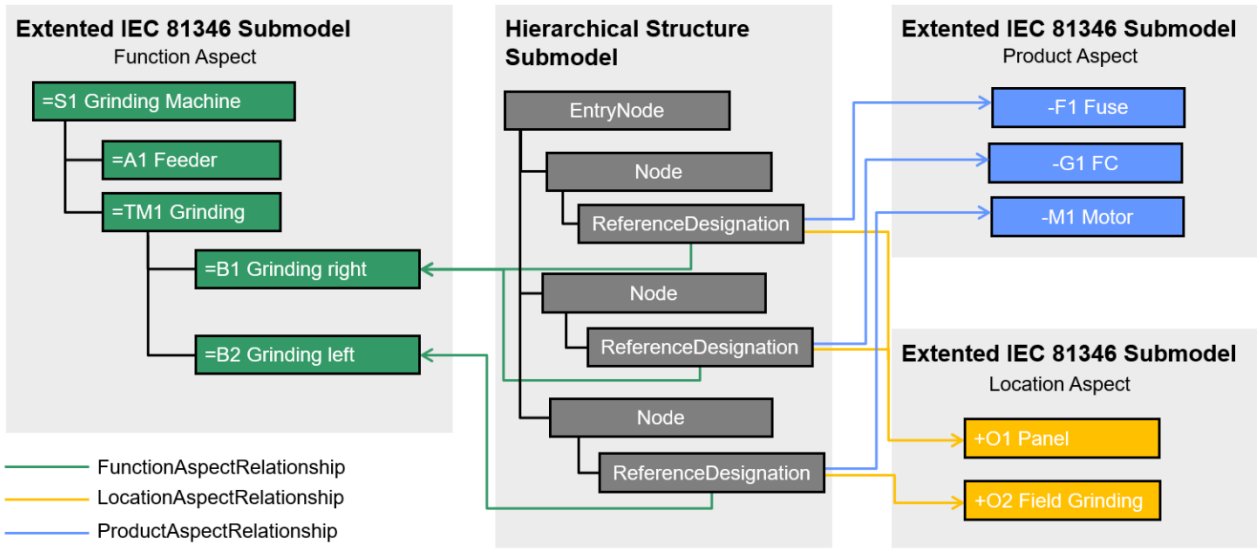

Figure 4 Example usage of this Submodel (IEC 81346 Submodel) with another Submodel (Hierachical Structure Submodel)

2.3 Attributes of the Submodel

For the Submodel, these attributes need to be set:

Table 4: Attribute of the Submodel

idShort:	IEC81346Structure Note: IdShort can be chosen freely considering its uniqness and the rules applied to NameType in AAS Part 1 V3. [6]		
Class:	Submodel – IEC81346Structure		
semanticId:	Constraint: The aspect implemented with the submodel must be uniquely described using the semanticIds listed below. Only the AspectNode that corresponds to the corresponding semanticId may be used. Function Aspect: [IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/FunctionAspect/1/0 ProductAspect: [IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/ProductAspect/1/0 LocationAspect: [IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/LocationAspect1/0 TypeAspect: [IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/TypeAspect/1/0 OtherAspect: [IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/OtherAspect/1/0 The model can also be described generically using supplementalSemanticId.		
supplementalSemanticId	[IRI] https://admin-shell.io/idta/SubmodelTemplate/HierarchicalStructures/ExtensionIEC81346/1/0		
Parent:	AAS		
Explanation:	Definition of the Submodel Extension IEC 81346 of HierarchicalStructures identified by its semanticId. The Submodel idShort can be picked freely.		
Inherits from:	HierarchicalStructures		
[SME type]	**semanticId = [idType]value**	**[valueType]**	**card.**
idShort	**Description@en**	**example**	
[Entity] IEC81346EntryNode	[IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/IEC81346EntryNode/1/0 Base entry point for the Entity tree in this Submodel, this must be a Self-managed Entity reflecting the Assets administrated in the AAS this Submodel is part of. Inherits from EntryNode	[-] -	1

Table 5: SubmodelElements of IEC81346EntryNode

idShort:	IEC81346EntryNode Note: IdShort can be chosen freely considering its uniqness and the rules applied to NameType in AAS Part 1 V3. [6]		
Class:	Entity - IEC81346EntryNode		
semanticId:	[IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/IEC81346/EntryNode/1/0		
Parent:	SM IEC81346Structure		
Explanation:	Base entry point for the Entity tree in this Submodel, this must be a Self-managed Entity reflecting the Assets administrated in the AAS this Submodel is part of. Inherits from EntryNode Note: The idShort of the EntryNode can be picked freely and may reflect a name of the asset.		
Inherits from:	HierarchicalStructures		
[SME type]	**semanticId = [idType]value**	**[valueType]**	**card.**
idShort	**Description@en**	**example**	
[Entity] IEC81346EntryNode	[IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/IEC81346/EntryNode/1/0 Base entry point for the Entity tree in this Submodel, this must be a Self-managed Entity reflecting the Assets administrated in the AAS this Submodel is part of. Inherits from EntryNode.	[-] -	1..*

2.4 SubmodelElements of IEC81346EntryNode

Table 6: SubmodelElements of IEC81346EntryNode

idShort:	IEC81346EntryNode Note: IdShort can be chosen freely considering its uniqness and the rules applied to NameType in AAS Part 1 V3. [6]		
Class:	Entity - IEC81346EntryNode		
semanticId:	[IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/IEC81346EntryNode/1/0		
Parent:	IEC81346EntryNode		
Explanation:	Base entry point for the Entity tree in this Submodel, this must be a Self-managed Entity reflecting the Assets administrated in the AAS this Submodel is part of. Inherits from EntryNode.		
Inherits from:	EntryNode		
[SME type]	**semanticId = [idType]value**	**[valueType]**	**card.**

idShort	Description@en	example	
[Entity] AspectNode	[IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/AspectNode/1/0 The Entity AspectNode can be a co-managed or self-managed entity representing an Aspect. This node is abstract and must be concretised by a concrete class as defined in Table 2. Note: IdShort can be chosen freely considering its uniqness and the rules applied to NameType in AAS Part 1 V3. [6]	[-] FunctionAspectNode or ProductAspectNode	1..*
[Property] PrefixLevel	[IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/PrefixLevel/1/0 PrefixLevel for the precise differentiation of an aspect and its characteristics. The appropriate AspectNode must be selected based on this prefix. Example: In electrical design, the functional assignment '==' and the function identifier '='	=, ==, ===	1
[Property] HierarchyPrefix	[IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/HierarchyPrefix/1/0 HierarchyPrefix to define the representation of the reference designator. Example: =TM1.B1	.	0..1
[Rel] HasStructure	[IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/hasStructure/1/0 Modeling of logical connections between components and sub-components. Either this or "IsPartOf" must be used, not both.		0..*

2.5 SubmodelElements of AspectNode

Table 7: SubmodelElements of AspectNode

idShort:	AspectNode <<abstract>>		
Class:	Entity - AspectNode		
semanticId:	[IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/AspectNode/1/0		
Parent:	IEC81346EntryNode		
Explanation:	Abstract class that describes an aspect in more detail. The semantic detailing is done in the derived classes, see Table 2. A Node reflects an element in the structure, with its sub aspects. The name of a node can be picked freely but it must be unique in its hierarchical (sub-)level.		
Inherits from:	Node		
[SME type]	**semanticId = [idType]value**	**[valueType]**	**card.**
idShort	**Description@en**	**example**	

[Property] LetterCodeLevel_1	[IRDI] 0173-1#02-AAC314#005 Used to classify objects based on code letters across three levels. The first letter represents the class (example: B), the two letters represent the sub-class (example: BA), the three letters represent the sub-sub-class (example: BAB) based on the applicable standard. See also 1.4.2	M	0..1
[Property] LetterCodeLevel_2	[IRDI] 0173-1#02-AAC341#005 Used to classify objects based on code letters across three levels. The two letters represent the sub-class (example: BA) based on the applicable standard. See also 1.4.2	A	0..1
[Property] LetterCodeLevel_3	[IRDI] 0173-1#02-ABC244#002 Used to classify objects based on code letters across three levels. The third level represents the sub-sub-class (example: BAB) based on the applicable standard. See also 1.4.2	A	0..1
[Property] AspectNumber	[IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/AspectNumber/1/0 Serves to further refine the classes, if the same letter codes are assigned in a system, the number is incremented in each case. (e.g.: BAB1, BAB2, ...). This achieves uniqueness in a system. See also 1.4.2	123	0..1
[MultiLangString] AspectDescription	[IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/AspectDescription/1/0 Brief description of the aspect in question. The description should be self-explanatory, as it also serves to define the 'OtherAspect'.	Transport infeed	0..1

2.5.1 SubmodelElements of specific Aspect Node

Table 8:Additional definitions of FunctionAspectNode

idShort:	FunctionAspectNode Note: IdShort can be chosen freely considering its uniqness and the rules applied to NameType in AAS Part 1 V3. [6] Recommandation: FunctionAspect__00__
Class:	Entity - FunctionAspectNode
semanticId:	[IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/FunctionAspectNode/1/0
Parent:	IEC81346EntryNode
Explanation:	Can be a Co-managed or Self-managed entity. A Node reflects an element in the hierarchical model is set into relation with one or more defined relations. The name of a node can be picked freely but it must be unique in its hierarchical (sub-)level.

Inherits from:	AspectNode		
[SME type]	**semanticId = [idType]value**	**[valueType]**	**card.**
idShort	**Description@en**	**example**	
[Rel] FullFilledBy	[IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/FullFilledBy/1/0 Constraint SM_IEC81346_001: Only allowed to point to a ProductAspectNode.	[-] -	0..*

Table 9: Additional definitions of ProductAspectNode

idShort:	ProductAspectNode Note: IdShort can be chosen freely considering its uniqness and the rules applied to NameType in AAS Part 1 V3. [6] Recommandation: ProductAspect__00__		
Class:	Entity - ProductAspectNode		
semanticId:	[IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/ProductAspectNode/1/0		
Parent:	IEC81346EntryNode		
Explanation:	Can be a Co-managed or Self-managed entity. A Node reflects an element in the hierarchical model is set into relation with one or more defined relations. The name of a node can be picked freely but it must be unique in its hierarchical (sub-)level.		
Inherits from:	AspectNode		
[SME type]	**semanticId = [idType]value**	**[valueType]**	**card.**
idShort	**Description@en**	**example**	
[Rel] LocatedAt	[IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/LocatedAt/1/0 Constraint SM_IEC81346_002: Only allowed to point to a LocationAspectNode.	[-] -	0..*
[Rel] FullFills	[IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/FullFills/1/0 Constraint SM_IEC81346_003: Only allowed to point to a FunctionAspectNode.	[-] -	0..*

Table 10: Additional definitions of LocationAspectNode

idShort:	LocationAspectNode Note: IdShort can be chosen freely considering its uniqness and the rules applied to NameType in AAS Part 1 V3. [6] Recommandation: LocationAspect__00__
Class:	Entity - LocationAspectNode
semanticId:	[IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/LocationAspectNode/1/0
Parent:	IEC81346EntryNode
Explanation:	Can be a Co-managed or Self-managed entity. A Node reflects an element in the hierarchical model is set into relation with one or more defined relations. The name of a node can be picked freely but it must be unique in its hierarchical (sub-)level.
Inherits from:	AspectNode

Table 11: Additional definitions of TypeAspectNode

idShort:	TypeAspectNode Note: IdShort can be chosen freely considering its uniqness and the rules applied to NameType in AAS Part 1 V3. [6] Recommandation: TypeAspect__00__
Class:	Entity - TypeAspectNode
semanticId:	[IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/TypeAspectNode/1/0
Parent:	IEC81346EntryNode
Explanation:	Can be a Co-managed or Self-managed entity. A Node reflects an element in the hierarchical model is set into relation with one or more defined relations. The name of a node can be picked freely but it must be unique in its hierarchical (sub-)level.
Inherits from:	AspectNode

Table 12: Additional definitions of OtherspectNode

idShort:	OtherAspectNode Note: IdShort can be chosen freely considering its uniqness and the rules applied to NameType in AAS Part 1 V3. [6] Recommandation: OtherAspect__00__
Class:	Entity - OtherAspectNode
semanticId:	[IRI] https://admin-shell.io/idta/HierarchicalStructures/ExtensionIEC81346/OtherAspectNode/1/0
Parent:	IEC81346EntryNode

Explanation:	Can be a Co-managed or Self-managed entity. A Node reflects an element in the hierarchical model is set into relation with one or more defined relations. The name of a node can be picked freely but it must be unique in its hierarchical (sub-)level.
Inherits from:	AspectNode

Annex A. Explanations on used table formats

1. General

The used tables in this document try to outline information as concise as possible. They do not convey all information on Submodels and SubmodelElements. For this purpose, the definitive definitions are given by a separate file in form of an AASX file of the Submodel Template and its elements.

2. Tables on Submodels and SubmodelElements

For clarity and brevity, a set of rules is used for the tables for describing Submodels and SubmodelElements.

- The tables follow in principle the same conventions as in [5].
- The table heads abbreviate 'cardinality' with 'card'.
- The tables often place two informations in different rows of the same table cell. In this case, the first information is marked out by sharp brackets [] form the second information. A special case are the semanticIds, which are marked out by the format: (type)(local)[idType]value.
- The types of SubmodelElements are abbreviated:

SME type	SubmodelElement type
Property	Property
MLP	MultiLanguageProperty
Range	Range
File	File
Blob	Blob
Ref	ReferenceElement
Rel	RelationshipElement
SMC	SubmodelElementCollection

- If an idShort ends with '__00__', this indicates a suffix of the respective length (here: 2) of decimal digits, in order to make the idShort unique. A different idShort might be choosen, as long as it is unique in the parent's context.
- The Keys of semanticId in the main section feature only idType and value, such as: [IRI]https://admin-shell.io/vdi/2770/1/0/DocumentId/Id. The attributes "type" and "local" (typically "ConceptDescription" and "(local)" or "GlobalReference" and (no-local)") need to be set accordingly; see [6].
- If a table does not contain a column with "parent" heading, all represented attributes share the same parent. This parent is denoted in the head of the table.
- Multi-language strings are represented by the text value, followed by '@'-character and the ISO 639 language code: example@EN.
- The [valueType] is only given for Properties.

Annex B. Changes to the Submodel template

General

This annex lists the changes from version to version of the Submodel, together with major changes in the overall document. Non-backward compatible changes (nc) are marked as such.

nc="x" means non-backward compatible; if no value is added in the table, then the change is backward compatible.

nc="(x)" means that the change made was implicitly contained or stated in the document before and is now being formalized. Therefore, the change is considered to be backward compatible.

Three tables are introduced to explain the changes:

1. changes with respect to previous version,
2. new elements in metamodel w.r.t previous version,
3. new, changed, or removed constraints w.r.t previous version.

If there are no changes the corresponding tables are omitted.

Annex C. Acknowledgment

The author thanks the International Electrotechnical Commission (IEC) for permission to reproduce Information from its International Standards. All such extracts are copyright of IEC, Geneva, Switzerland. All rights reserved. Further information on the IEC is available from www.iec.ch. IEC has no responsibility for the placement and context in which the extracts and contents are reproduced by the author, nor is IEC in any way responsible for the other content or accuracy therein.

Bibliography

[1] "Recommendations for implementing the strategic initiative INDUSTRIE 4.0", acatech, April 2013. [Online]. Available https://www.acatech.de/Publikation/recommendations-for-implementing-the-strategic-initiative-industrie-4-0-final-report-of-the-industrie-4-0-working-group/

[2] "Implementation Strategy Industrie 4.0: Report on the results of the Industrie 4.0 Platform"; BITKOM e.V. / VDMA e.V., /ZVEI e.V., April 2015. [Online]. Available: https://www.bitkom.org/noindex/Publikationen/2016/Sonstiges/Implementation-Strategy-Industrie-40/2016-01-Implementation-Strategy-Industrie40.pdf

[3] "The Structure of the Administration Shell: TRILATERAL PERSPECTIVES from France, Italy and Germany", March 2018, [Online]. Available: https://www.plattform-i40.de/I40/Redaktion/EN/Downloads/Publikation/hm-2018-trilaterale-coop.html

[4] "Beispiele zur Verwaltungsschale der Industrie 4.0-Komponente – Basisteil (German)"; ZVEI e.V., Whitepaper, November 2016. [Online]. Available: https://www.zvei.org/presse-medien/publikationen/beispiele-zur-verwaltungsschale-der-industrie-40-komponente-basisteil/

[5] "Verwaltungsschale in der Praxis. Wie definiere ich Teilmodelle, beispielhafte Teilmodelle und Interaktion zwischen Verwaltungsschalen (in German)", Version 1.0, April 2019, Plattform Industrie 4.0 in Kooperation mit VDE GMA Fachausschuss 7.20, Federal Ministry for Economic Affairs and Energy (BMWi), Available: https://www.plattform-i40.de/PI40/Redaktion/DE/Downloads/Publikation/2019-verwaltungsschale-in-der-praxis.html

[6] "Specification of the Asset Administration Shell; Part 1: Metamodel – IDTA Number 01001-3-0, April 2023, [Online]. Available: https://industrialdigitaltwin.org/wp-content/uploads/2023/06/IDTA-01001-3-0_SpecificationAssetAdministrationShell_Part1_Metamodel.pdf

[7] "Details of the Asset Administration Shell; Part 2: Application Programming Interfaces – IDTA Number 01002-3-0", June 2023, [Online]. Available: https://industrialdigitaltwin.org/wp-content/uploads/2023/06/IDTA-01002-3-0_SpecificationAssetAdministrationShell_Part2_API_.pdf

[8] Industrielle Systeme, Anlagen und Ausrüstungen und Industrieprodukte – Strukturierungsprinzipien und Referenzkennzeichnung – Teil 1: Allgemeine Regeln (2009) (IEC 81346-1:2009)

[9] "OPC 10000-3: UA Part 3: Address Space Model - NonHierarchicalReferences ReferenceType" , Dezember 2024 [Online] UA Part 3: Address Space Model - 7.4 NonHierarchicalReferences ReferenceType

www.industrialdigitaltwin.org