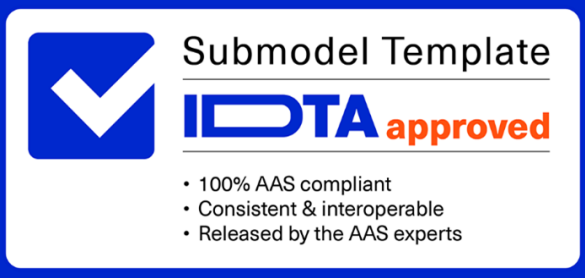
IDTA 02099-1: Digital Product Passport - Part 1

Metadata

May 2026

SPECIFICATION

Submodel Template of the Asset Administration Shell



IDTA 02099-1

Imprint

Publisher

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

Version history

Date	Version	Comment
28.05.2026	1.0	Release of the official Submodel template published by IDTA.

Table of Contents

Chapter 1. General

1.1. About this document

This document is a part of an overall specification series [4]. Each part specifies the contents of a Submodel Template (SMT). The specifications of the Asset Administration Shell (AAS) are the basis for the Submodel Template specifications, see [3].

The target audience of the specification are developers and editors of technical documentation and manufacturer information, which are describing assets 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 specification was created following the "semantic-driven workflow" as defined in [5] based on Aspect Models [6].

In this specification the following sources are used for defining semantics:

- Aspect Models published at IDTA [7]: https://github.com/admin-shell-io/smt-semantic-models, models with namespace "io.admin-shell"

1.2. Scope of the Submodel

This Submodel template aims to define the data points that are required for every Digital Product Passport conformant to the specification of EN 18223:2026 [1].

This document represents the first part in a planned series. It is envisioned that future parts will be developed to address additional aspects; however, the scope, content, and timeline for these subsequent publications have not yet been determined.

EN 18223 is part of a series of standards pepared by the Joint Technical Committee JTC24 of CEN and CENELEC:

- EN 18223 - Module 4: Interoperability (JSON Payload)
- EN 18222 - Module 8: API
- EN 18239 - Module 3: "Digital Product Passport - access rights management, information system security, and business confidentiality
- EN 18219 - Module 1: Unique identifiers
- EN 18221 - Module 6: Data Storage, archiving etc.
- EN 18216 - Module 5: Data Processing, data exchange
- EN 18220 - Module 2: Data Carriers and digital representation
- EN 18246 - Module 7: Data authentication, reliability and integrity

In this document they are referred to as the "JTC24 specifications".

1.3. Relevant standards for the Submodel Template

This Submodel template fulfills the requirements for data points as defined in CEN CENELEC JTC24 standard EN 18223:2026.

1.4. Explanations on used UML diagrams

For clarity and an improved legibility readers suggested to go through this section at first before reading the following chapters.

UML diagrams feature box-like elements, called "classes". These classes, typically Submodels, SubmodelElementCollections or SubmodelElementLists, typically feature a set of Properties or further SubmodelElements. These elements can have specific cardinalities.

The single classes are hierarchally organized by aggregation relations, these can be seen as "contains" relation.

For a further overview on UML diagrams please refer to [2] and [3].

For further details about used table formats please refer to Chapter 4.

Chapter 2. Information set for Submodel “Digital Product Passport - Metadata”

2.1. General

A Digital Product Passport (DPP) is a digital record of product characteristics throughout its life cycle [1]. Example characteristics include environmental sustainability, environmental impact, and recyclability [1].

Figure 1 shows exemplary for the Battery Passport how a DPP is created based on the Asset Administration Shell (AAS): 7 standardized Submodel Template Specifications were created (based on DIN DKE SPEC 91100) to represent the Battery Passport in total (series IDTA-02035). At the point of publication of the requirements for Battery Passport EN 18223 were not yet published.

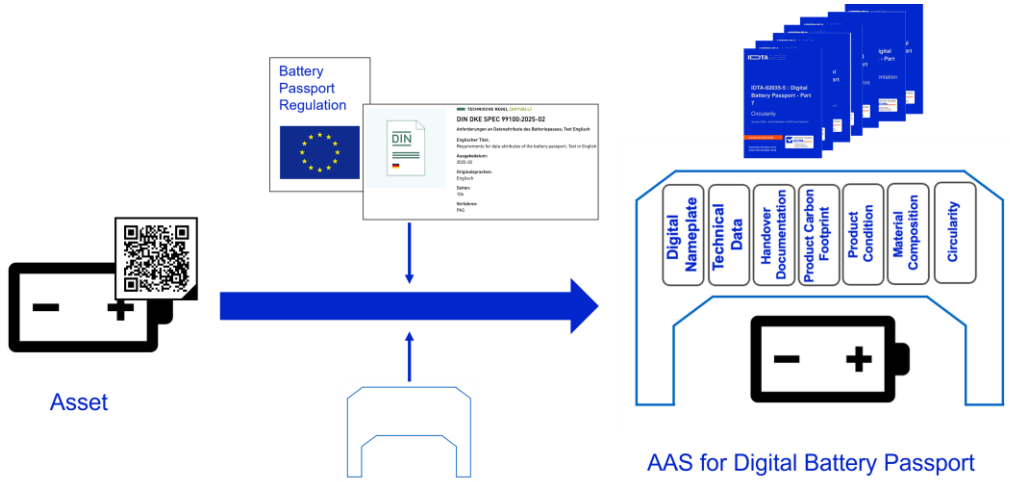


Figure 1. Example: AAS for Digital Battery Passport

To ensure harmonization across different implementations, this document describes a single, recommended approach for creating a Digital Product Passport (DPP) using an Asset Administration Shell (AAS) as specified in IDTA-01001 and IDTA-01002. While other methods may exist, this document focuses exclusively on this proposed pathway to establish a consistent and interoperable standard between the DPP Metadata Submodel and the other Submodels needed to compose a DPP.

It is important to understand that an Asset Administration Shell (AAS) is not, by itself, the Digital Product Passport as required by EN 18222 or EN 18223. However, a DPP can be easily derived from an existing AAS. The prerequisite is that the AAS contains the Submodels supporting the regulation’s required data points for the product, as well as the DPP metadata Submodel specified in this document.

The Digital Battery Passport provides a clear example of this relationship. It is represented by seven standardized Submodel specifications (see Figure 2). To create the complete passport, these Submodels are merged with the DPP metadata Submodel defined herein, resulting in a comprehensive Digital Product Passport conformant to regulation.

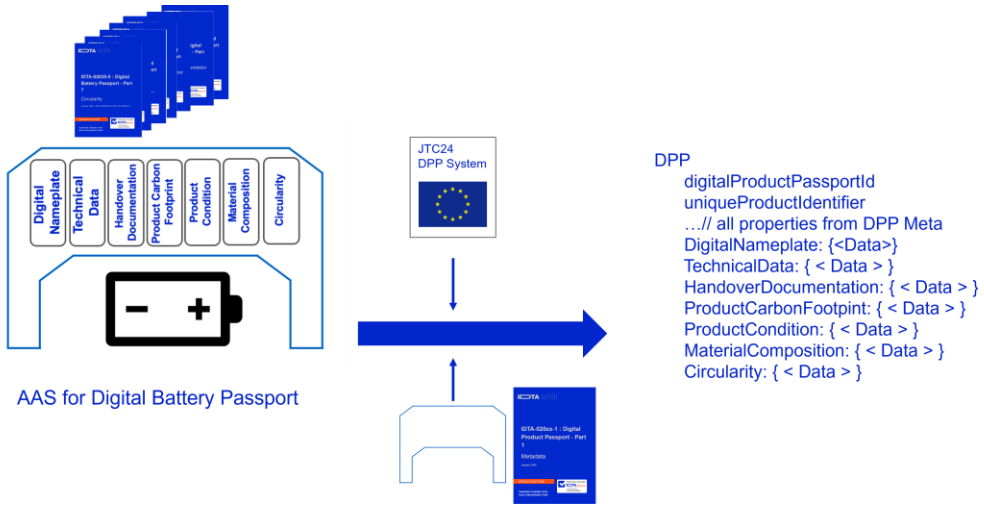


Figure 2. Example: DPP derived from AAS

The primary advantage of using an Asset Administration Shell (AAS) to implement a Digital Product Passport (DPP) is its modular design, which contrasts with the monolithic structure assumed by the JTC24 specification.

An AAS acts as a container for multiple, independent Submodels. Each Submodel has a unique identifier and can be managed and reused separately. This modularity offers significant benefits:

- Data Reuse: Information common to many products can be defined once and reused across countless instances.
- Simplified Access Management: Permissions can be set for individual Submodels.
- Efficient Data Onboarding: New information can be added as a new Submodel without altering the existing structure.

The Battery Passport provides a perfect example. While the regulation requires a passport for every battery, much of the data is identical across batteries of the same type or from the same production facility. With an AAS, you can create a hierarchy of information, where the AAS for a single battery instance reuses Submodels from the AAS of its production facility and its battery type. This same principle allows for standardized Submodels, like contact information, to be reused across entirely different domains.

While the AAS approach is modular, it is designed to produce the single, monolithic JSON document that JTC24 expects. The key is understanding that even this single document must adhere to a set of different schemas. For a DPP to be compliant, it must be validated against at least two types:

- The overall generic schema implicitly defined by JTC24 (derived from EN 18223).
- The specific schema derived from for the relevant vertical regulation (e.g., the Battery Passport "delegated act").
- Specific schemas derived from relevant horizontal regulations.

The AAS framework directly supports this structure. It realizes this by registering new Submodels to an AAS and then adding the payload of those Submodels to the payload of the overall DPP. In essence, the AAS dynamically assembles the required monolithic document by aggregating the data from its various registered Submodels. This allows you to benefit from a modular development and management process while still delivering a compliant, single-file DPP that satisfies all schema requirements.

The process of generating the DPP from an AAS through a sequence of operation calls is detailed in the following subsections and illustrated in Figure 3.

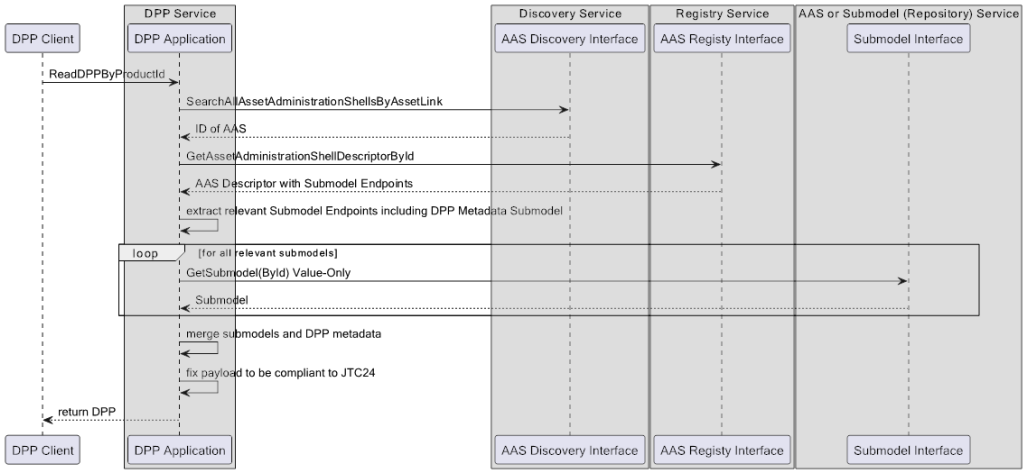


Figure 3. Example: Sequence of Operation Calls and Steps

The sequence diagram included herein is provided for informative purposes only, offering a contextual example to aid in understanding; it should not be considered an exhaustive or normative representation of all possible interactions. For implementation details, readers are directed to the metamodel in Part 1 and the API in Part 2 of the Asset Administration Shell specifications [3].

2.1.1. Data Points of every single DPP

In Example DPPMetadata JSON Payload an example JSON instance of this Submodel Template Specification is shown. It describes a DPP using Submodel Template semanticIds as contentSpecificationIds, here (example) semanticIds for Nameplate, TechnicalData, HandoverDocumentation, ProductCondition, MaterialComposition and Circularity for a specific Product Passport.

Note: While standard EN 18223 provides an illustrative example for the contentSpecificationId ("prEN1234_xyz"), it is important to recognize that this format is forward-looking. To date, no formal standards for vertical-specific digital product passports have been specified. Therefore, in the absence of a definitive specification, this document puts forth a recommendation to ensure a harmonized approach.

Example DPPMetadata JSON Payload

```
{
  "digitalProductPassportId" : "https://www.example.org/dpp/1234545",
  "uniqueProductIdentifier": "https://www.example.org/1234545",
  "granularity": "Item",
  "dppSchemaVersion": "ENXXX:v1.0",
  "dppStatus": "Active",
  "lastUpdate": "2025-08-22T03:12:00Z",
  "economicOperatorId": "gxx:ppp456789",
  "facilityId": "gxx:xxx987654",
  "contentSpecificationIds": [
      "https://admin-shell-io/idta/digitalpassport/Nameplate/1",
      "https://admin-shell-io/idta/digitalpassport/TechnicalData/1"},
      "https://admin-shell-io/idta/digitalpassport/HandoverDocumentation/2",
      "https://admin-shell-io/idta/digitalpassport/ProductCondition/1",
      "https://admin-shell-io/idta/digitalpassport/MaterialComposition/1",
      "https://admin-shell-io/idta/digitalproductpassport/Circularity/1"
    ]
}
```

Although some attributes (see Example DPP JSON Payload) of this Submodel specification can be mapped to attributes in the Asset Administration Shell (AAS) metamodel, the formal recommendation is to instead use this predefined Submodel Template IDTA-02099 for this metadata. This template is derived from the SMT IDTA-02006-4-0 Nameplate. For the corresponding semantic identifiers, please refer to their respective specifications.

The payload required by EN 18223 for the DPP metadata maps to the Value-Only format of this Submodel.

Note: for the other Submodels used to build a DPP EN 18223 allows two different formats "compressed" and "explanded" with compressed being mandatory to be supported. For DPP Metadata there is exactly one correct serialization since it is part of the overall schema and not of a specific content.

Table 1. Mapping JTC24 Meta Data to AAS

EN 18223	**AAS**	**IDTA-02006-3-0**	**comment**
digitalProductPassportId	AssetAdministrationShell/id	—	Although the ID of the AAS is very similar to the DPP ID they are not necessarily the same because the DPP is a monolithic output whereas the ID of an AAS does not imply a specific set of data in a specific point in time because the Submodels are identifiable
uniqueProductIdentifier	AssetInformation/globalAssetId	URIOfTheProduct	Identifier types allowed are specified in EN 18219
granularity	assetKind	—	for mapping of enumeration values see Table 2
dppSchemaVersion	—		In EN 1822:2026 no Schema is defined and thus there is also no schema version.
dppStatus	—	—	In EN 1822:2026 no enumeration is specified for the Status, only example values are given like "Active". The AAS does not support administrative information on its status. However, status of the AAS does not need to be identical to the status of the DPP.
lastUpdate	AdministrativeInformation/updatedAt	—	EN 18223 does not support explicit versioning (like the AAS does), versioning is supported via the "lastUpdate" attribute in DPPs
economicOperatorId	—	—	Identifier types allowed are specified in EN 18219
facilityId	—	UniqueFacilityIdentifier	Identifier types allowed are specified in EN 18219

contentSpecificationIds	—		This attribute reminds of AssetAdministrationShell/submodels because it lists the data sets relevant for the different DPP(s). A DPP may contain data from several delegated acts, i.e. from different regulations.

Table 2. Mapping EN 18223 granularity to AAS AssetKind

EN 18223 granularity	**AAS AssetKind**	**comment**
Item	Instance	
Model	Type	AAS supports the reuse of Submodels specified on Type Level also on Batch or Instance Level. EN 18223 does not support reuse via its metamodel: resue is to be taken care of by the Economic Operator or Service Provider.
Batch	Batch	
—	Role	no equivalent in ESPR
—	NotApplicable	not foreseen in ESPR

2.1.2. Creating of DPP with several Submodels

Given the example data for the DPP in Example DPPMetadata JSON Payload the example is now extended with the data of the corresponding Submodels itself (see Example DPP JSON Payload). The JSON element name corresponds directly to the contentSpecificationId. It is the starting point for the corresponding Submodel data.

Example DPP JSON Payload

```
{
  "digitalProductPassportId":
    "https://www.example.org/dpp/1234545",

  "uniqueProductIdentifier":
    "https://www.example.org/1234545",

  "granularity": "Item",
  "dppSchemaVersion": "ENXXX:v1.0",
  "dppStatus": "Active",
  "lastUpdate": "2025-08-22T03:12:00Z",
  "economicOperatorId": "gxx:ppp456789",
  "facilityId": "gxx:xxx987654",

  "contentSpecificationIds": [
    "https://admin-shell-io/idta/digitalproductpassport/Nameplate/1",
    "https://admin-shell-io/idta/digitalproductpassport/TechnicalData/1",
    "https://admin-shell-
io/idta/digitalproductpassport/HandoverDocumentation/2",
    "https://admin-shell-io/idta/digitalproductpassport/ProductCondition/1",
    "https://admin-shell-
io/idta/digitalproductpassport/MaterialComposition/1",
    "https://admin-shell-io/idta/digitalproductpassport/Circularity/1"
  ],

  "https://admin-shell-io/idta/digitalproductpassport/Nameplate/1": {
    "<Data>": {}
  },

  "https://admin-shell-io/idta/digitalproductpassport/TechnicalData/1": {
    "<Data>": {}
  },

  "https://admin-shell-
io/idta/digitalproductpassport/HandoverDocumentation/2": {
    "<Data>": {}
  },

  "https://admin-shell-io/idta/digitalproductpassport/ProductCondition/1": {
    "<Data>": {}
  },

  "https://admin-shell-
io/idta/digitalproductpassport/MaterialComposition/1": {
    "<Data>": {}
  },

  "https://admin-shell-io/idta/digitalproductpassport/Circularity/1": {
    "<Data>": {}
  }
}
```

In Example AAS Value-Only JSON Payload for Battery Passport, PCF data an example JSON payload conformant to IDTA-02023 Version 1.0 Carbon Footprint (see CarbonFootprintBattery.json) in Value-Only format is provided. In this case no multi language properties or File elements are contained and this is why this payload is also conformant to the compressed format as specified in EN 18223. CarbonFootprintBattery-schema.json, generated from the Aspect Model conformant to IDTA-01023, is the schema that can be combined with the one of the EN 18223 generic schema.

Example AAS Value-Only JSON Payload for Battery Passport, PCF data

```
{
  "ProductCarbonFootprints" : [ {
    "QuantityOfMeasureForCalculation" : 5.0,
    "PcfCo2eq" : 17.2,
    "ReferenceImpactUnitForCalculation" :
    "g",
    "WebLinkToPublicCarbonFootprintStudy" : [ "eOMtThyhVNLWUZNRcBaQK" ],
    "PcfCalculationMethods" : [ "ISO 14067" ],
    "PerformanceClass" : "eOMtThyhVNLWUZNRcBaQKxI",
    "LifeCyclePhases" : [ "C4 - landfill" ]
  } ]
}
```

So in combination it would be the "semanticId" of SMT IDTA-02035-3 that is "https://admin-shell.io/idta/CarbonFootprint/CarbonFootprint/1/0", (see Example DPP compressed serialization for Battery Passport, PCF data). The second value to be standardized is the one for the contentSpecificationIds. For details on "contentSpecificationIds" see section 3.2.

Example DPP compressed serialization for Battery Passport, PCF data

```
{
 "digitalProductPassportId": "https://www.example.org/batterypassport/1234545",
 "uniqueProductIdentifier": "https://www.example.org/1234545",
 "granularity": "Item", "dppSchemaVersion":
 "ENXXX:v1.0", "dppStatus": "Active",
 "lastUpdate": "2025-08-22T03:12:00Z",
 "economicOperatorId": "gxx:ppp456789",
 "facilityId": "gxx:xxx987654",
 "contentSpecificationIds": [
    "https://admin-shell.io/idta/CarbonFootprint/CarbonFootprint/1/0"
  ],
 "https://admin-shell.io/idta/CarbonFootprint/CarbonFootprint/1/0" :
    {
      "ProductCarbonFootprints" : [ {
        "QuantityOfMeasureForCalculation" : 5.0,
        "PcfCo2eq" : 17.2, "ReferenceImpactUnitForCalculation"
        : "g",
        "WebLinkToPublicCarbonFootprintStudy" : [ "eOMtThyhVNLWUZNRcBaQK" ],
        "PcfCalculationMethods" : [ "ISO 14067" ],
        "PerformanceClass" : "eOMtThyhVNLWUZNRcBaQKxI",
        "LifeCyclePhases" : [ "C4 - landfill" ]
      } ]
    }
}
```

In Example JTC24 expanded Full serialization JSON Paylaod for Battery Passport, PCF data an example is shown how a payload similar to the AAS "Normal"-Format could look like.

Example JTC24 expanded Full serialization JSON Paylaod for Battery Passport, PCF data

```
{
 "digitalProductPassportId": "https://www.example.org/batterypassport/1234545",
 "uniqueProductIdentifier": "https://www.example.org/1234545",
 "granularity": "Item",
 "dppSchemaVersion": "ENXXX:v1.0",
 "dppStatus": "Active",
 "lastUpdate": "2025-08-22T03:12:00Z",
 "economicOperatorId": "gxx:ppp456789",
 "facilityId": "gxx:xxx987654",
    "https://admin-shell.io/idta/CarbonFootprint/CarbonFootprint/1/0"
  ],
 "https://admin-shell.io/idta/CarbonFootprint/CarbonFootprint/1/0" :
  [
   <...>
   {
    "elementId": "QuantityOfMeasureForCalculation",
    "objectType": "SingleValuedDataElement",
    "dictionaryReference": "https:/dictionary1.eu/QuantityOfMeasureForCalculation",
    "valueDataType": "xsd:float",
    "value": 5.0
    }
    <...>
  ]
}
```

For details on DPP serialization (compressed and expanded full) compared to Value-Only and Normal serialization of the AAS see IDTA-01001 Asset Administration Shell Specifications, Part 1.

2.2. Overview UML model

The SubmodelElements described in section 3 are structured in the following way (see Figure 4):

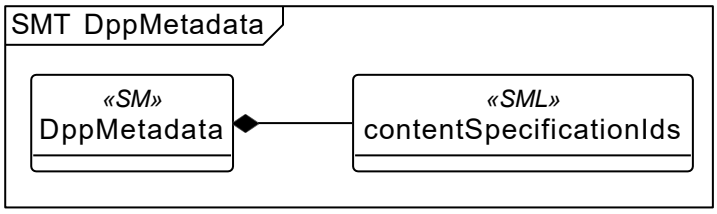


Figure 4. UML overview

Chapter 3. Information structures and attributes

3.1. Properties of the Submodel “DppMetadata”

Figure 5 shows the UML-diagram defining the relevant properties which need to be set.

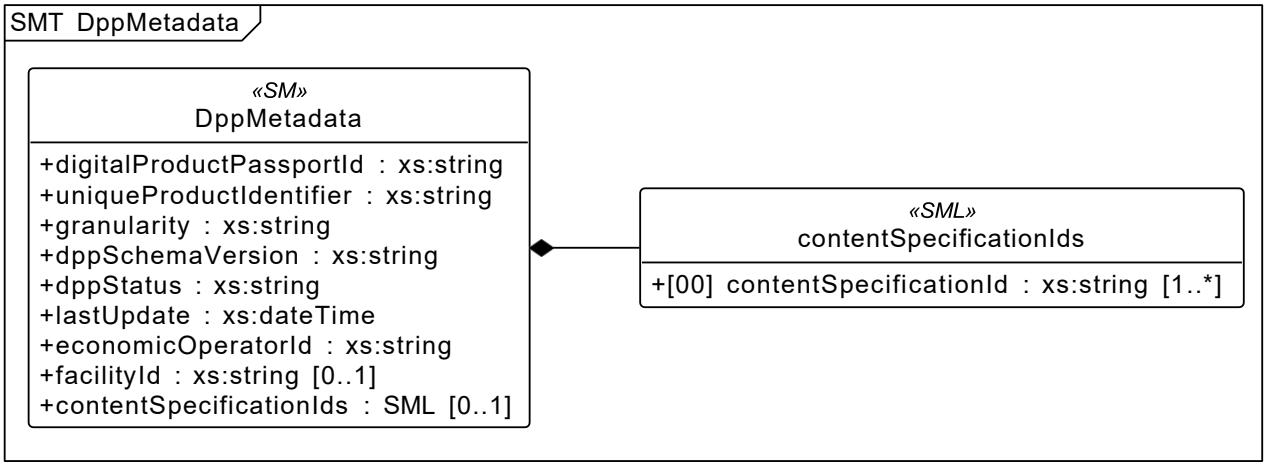


Figure 5. UML-Diagram for Submodel "DppMetadata"

For granularity three values are predefined, Model, Item and Batch.

Enumeration:	*GranularityEnum*
Explanation:	Enumeration
Set of:	—
semantic ID:	https://admin-shell.io/idta/cds/granularityValues/1
supplemental Semantic ID:	https://admin-shell.io/aas/3/2/AssetKind
Literal	**semantic ID**
	supplemental Semantic ID
	Explanation
Model	https://admin-shell.io/idta/cds/model/1
	https://admin-shell.io/aas/3/2/AssetKind/Type
	Model Note: Model is conformant to "Type" in "AssetKind" in the Asset Administration Shell Specification. The two values shall map.
Item	https://admin-shell.io/idta/cds/item/1
	https://admin-shell.io/aas/3/2/AssetKind/Instance
	Item Note: Item is conformant to "Instance" in "AssetKind" in the Asset Administration Shell Specification. The two values shall map.

Batch	`https://admin-shell.io/idta/cds/batch/1`
	`https://admin-shell.io/aas/3/2/AssetKind/Batch`
	Batch Note: Batch is conformant to "Batch" in "AssetKind" in the Asset Administration Shell Specification. The two values shall map.

There is no schema defined so far in EN 18223, so there also is no schema version. There also is no regular expression defined how a schema version should be specified.

idShort:	**DppMetadata**		
Class:	Submodel		
semanticId:	https://admin-shell.io/idta/cds/dppMetadata/1		
Parent:	Asset Administration Shell		
Explanation:	Digital record of product characteristics throughout its life cycle. SOURCE: EN 18223:2026(en)		
Element details:	-		
[SME type] idShort	semanticId Description@en	[valueType] example	card.
[Prop] digitalProductPassportId	https://admin-shell.io/idta/cds/digitalProductPassportId/1 **IsCaseOf:** • urn:samm:io.admin-shell.idta.dpp.dpp_metadata:1.0.0#digitalProductPassportId Unique identifier of an instance of the digital product passport. SOURCE: EN 18223:2026(en)	[String] https://www.example.org/dpp/1234545	1
[Prop] uniqueProductIdentifier	https://admin-shell.io/idta/cds/uniqueProductIdentifier/1 **IsCaseOf:** • urn:samm:io.admin-shell.idta.dpp.dpp_metadata:1.0.0#uniqueProductIdentifier • 0112/2///61360_7#CBA055#001 • 0173-1#02-ABH173#003 • urn:samm:io.admin-shell.idta.digital_nameplate:3.0.0#uriOfTheProduct • 0112/2///61987#ABN590#002 Unique string of characters for the identification of a product, that also enables a web link to the digital product passport. String formatted according to EN 18219:2026. SOURCE: EN 18223:2026(en)	[String] https://www.example.org/1234545	1

[Prop] granularity	https://admin-shell.io/idta/cds/granularity/1 **IsCaseOf:** • urn:samm:io.admin-shell.idta.dpp.dpp_metadata:1.0.0#granularity supplementalSemanticId: https://admin-shell.io/aas/3/2/AssetKind The granularity of the Product. There are three values: ‘Model‘, ‘Item‘ and ‘Batch‘. The term ‘model’ usually refers to a version of a product of which all units share the same technical characteristics relevant for the ecodesign requirements and the same model identifier, the term ‘batch’ usually refers to a subset of a specific model composed of all products produced in a specific manufacturing plant at a specific moment in time and the term ‘item’ usually refers to a single unit of a model. SOURCE: ESPR p2024/1781, REGULATION (EU) 2024/1781 OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL of 13 June 2024 establishing a framework for the setting of ecodesign requirements for sustainable products, amending Directive (EU) 2020/1828 and Regulation (EU) 2023/1542 and repealing Directive 2009/125/EC. URI: OJ:L_202401781	[String] Item	1
[Prop] dppSchemaVersion	https://admin-shell.io/idta/cds/dppSchemaVersion/1 **IsCaseOf:** • urn:samm:io.admin-shell.idta.dpp.dpp_metadata:1.0.0#dppSchemaVersion Reference standard the DPP instance schema refers to. SOURCE: EN 18223:2026(en)	[String] ENXXX:v1.0	1
[Prop] dppStatus	https://admin-shell.io/idta/cds/dppStatus/1 **IsCaseOf:** • urn:samm:io.admin-shell.idta.dpp.dpp_metadata:1.0.0#dppStatus Status of the DPP instance as digital resource. SOURCE: EN 18223:2026(en)	[String] Active	1

[Prop] lastUpdate	https://admin-shell.io/idta/cds/lastUpdate/1 **IsCaseOf:** • urn:samm:io.admin-shell.idta.dpp.dpp_metadata:1.0.0#lastUpdate **supplementalSemanticId:** https://admin-shell.io/aas/3/2/AdministrativeInformation/updatedAt Date and time of the latest update to the DPP instance. String formatted as Timestamp UTC-based according to ISO 8601-1:2019/Amd1:2022. SOURCE: EN 18223:2026(en)	[DateTime] 2025-08-22T03:12:00Z	1
[Prop] economicOperatorId	https://admin-shell.io/idta/cds/economicOperatorId/1 **IsCaseOf:** • urn:samm:io.admin-shell.idta.dpp.dpp_metadata:1.0.0#economicOperatorId Unique string of characters for the identification of an actor involved in a product's life cycle. String formatted according to EN 18219:2026. SOURCE: EN 18223:2026(en)	[String] gxx:ppp456789	1
[Prop] facilityId	https://admin-shell.io/idta/cds/facilityId/1 **IsCaseOf:** • urn:samm:io.admin-shell.idta.dpp.dpp_metadata:1.0.0#facilityId • 0173-1#02-AAV646#003 • https://admin-shell.io/idta/nameplate/3/0/UniqueFacilityIdentifier Unique string of characters for the identification of locations or buildings involved in a product's value chain or used by actors involved in a product's life cycle. String formatted according to EN 18219:2026. SOURCE: EN 18223:2026(en)	[String] gxx:xxx987654	0..1

[SML] contentSpecificationIds	https://admin-shell.io/idta/cds/contentSpecificationIds/1 **IsCaseOf:** • urn:samm:io.admin-shell.idta.dpp.dpp_metadata:1.0.0#contentSpecificationIds Reference to horizontal or product-type related content specifications for the DPP. SOURCE: EN 18223:2026(en)	[] 1 elements	0..1

3.2. Properties of the SML "contentSpecificationIds"

Figure 6 shows the UML-diagram for SML contentSpecificationIds.

Figure 6. UML-Diagram for SML "contentSpecificationIds"

A content specification ID refers to the content of the DPP.

Two different cases need to be distinguished:

- **regulations** A product is not necessarily subject to a single regulation, but sometimes to multiple. In this case it is not foreseen that different DPPs are created, one for each regulation for the same product but only one containing all information from all regulations.
- **content** As in Battery Passport (see IDTA (IDTA-02035 series of Submodel template specifications) a Passport may consist of several Submodels. Only the combination of all of them (plus the general information as defined in this specification) would then define the complete Digital Product Passport.

There are no rules standardized in CEN CENELEC how exactly the content specification IDs will be handled. So the following recommendations hold for DPPs created on basis of AAS Submodels.

The recommendation is to just use the "semanticId" of the Submodel Template Specification that belongs to the DPP under consideration as the unique Identifier of the content. If more than one regulation is affected then all these semantic IDs are added as well.

```
<semanticId of the corresponding Submodel Template Specification>
```

Example:

```
"contentSpecificationIds": [
   "https://admin-shell.io/idta/CarbonFootprint/1/0",
   "urn:samm:io.admin-shell.idta.batterypass.circularity:1.0.0#Circularity"
 ]
```

Since there might be more than one Submodel with the same semanticId in the same AAS, it would always be the newest version of the corresponding Submodel that needs to be included in the DPP.

idShort:	contentSpecificationIds
Class:	SubmodelElementList
semanticId:	https://admin-shell.io/idta/cds/contentSpecificationIds/1
Parent:	DppMetadata

Explanation:	Reference to horizontal or product-type related content specifications for the DPP. (SOURCE: EN 18223:2026(en))		
Element details:	orderRelevant=No, typeValueListElement=Property, valueTypeListElement=xs:string		
[SME type] idShort	semanticId Description@en	[valueType] example	card.
[Prop] contentSpecificationId	https://admin-shell.io/idta/cds/contentSpecificationId/1 **IsCaseOf:** • urn:samm:io.admin-shell.idta.dpp.dpp_metadata:1.0.0#ContentSpecificationId Reference to horizontal or product-type related content specification for the DPP. SOURCE: EN 18223:2026(en)	[String] https://admin-shell-io/idta/digitalproductpassport/Circularity/1	1..*

If *idShort* are added to the elements within the SML "contentSpecificationIds" the recommendation is to use the *idShort* of the corresponding Submodel Template with the specified *semanticId*.

Annex A. Explanations on used table formats

4.1. 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.

4.2. 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 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.
- Using “IsCaseOf”: Each SubmodelElement is identified by a primary IDTA-assigned semanticId (https://admin-shell.io/idta/cds/...). Where applicable, IsCaseOf references are provided, which, as defined in the AAS metamodel [3], are references to external definitions the concept is compatible with or was derived from. These may include concepts from external semantic sources such as ECLASS, and IEC CDD, or other external dictionaries and standards.
- The types of SubmodelElements are abbreviated (see Table 3):

Table 3. Abbreviations for SubmodelElements

SME type	SubmodelElement type
Blob	Blob
Cap	Capability
Ent	Entity
Evt	Event
File	File
MLP	MultiLanguageProperty
Opr	Operation
Prop	Property
Range	Range
Ref	ReferenceElement
Rel	RelationshipElement
RelA	AnnotatedRelationshipElement
SMC	SubmodelElementCollection
SME	SubmodelElement
SML	SubmodelElementList

• 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.

Changes Version 1.0

- First Version based on EN 18223:2026 and EN 18222:2026

Bibliography

[1] EN 18223:2026 "Digital Product Passport - System interoperability". CEN CENELEC.

[2] "OMG Unified Modeling Language (OMG UML)", Formal/2017-12-05, Version 2.5.1. December 2018. [Online] Available: https://www.omg.org/spec/UML/

[3] "Specification of the Asset Administration Shell", Publisher: Industrial Digital Twin Association (IDTA). [Online]. Available: https://industrialdigitaltwin.org/en/content-hub/aasspecifications

[4] "Submodel Templates", Publisher: Industrial Digital Twin Association (IDTA). [Online]. Available: https://industrialdigitaltwin.org/en/content-hub/submodels

[5] "How-to create a Submodel Template Specification", Publisher: Industrial Digital Twin Association (IDTA). June 2025. V1.1. [Online]. Available: https://industrialdigitaltwin.org/en/wp-content/uploads/sites/2/2025/06/IDTA_How-to-write-a-SMT-v1.1.pdf

[6] "Semantic Aspect Meta Model (SAMM)", V2.2.0. [Online]. Available: https://eclipse-esmf.github.io/samm-specification/2.2.0/index.html

[7] "Semantic Aspect Models - smt-semantic-models", Publisher: Industrial Digital Twin Association (IDTA). [Online]. Available: https://github.com/admin-shell-io/smt-semantic-models

[8] EN 18222:2026 "Digital Product Passport - Application Programming Interfaces (APIs) for the product passport lifecycle management and searchability". CEN CENELEC.

www.industrialdigitaltwin.org