



DPC: 23 / 30459609 DC

BSI Group Headquarters

389 Chiswick High Road London W4 4AL

Tel: +44 (0)20 8996 9000

Fax: +44 (0)20 8996 7400

www.bsigroup.com

Date: 14 March 2023

Origin: International

Latest date for receipt of comments: 3 May 2023

Project No. 2022/01732

Responsible committee: GEL/65 Measurement and control

Interested committees:

Title: BS EN IEC 63278-2 Asset Administration Shell for Industrial Applications

Part 2: Information meta model

Please notify the secretary if you are aware of any keywords that might assist in classifying or identifying the standard or if the content of this standard

- i) has any issues related to 3rd party IPR, patent or copyright
- ii) affects other national standard(s)
- iii) requires additional national guidance or information

**WARNING: THIS IS A DRAFT AND MUST NOT BE REGARDED OR USED AS A BRITISH STANDARD.
THIS DRAFT IS NOT CURRENT BEYOND 3 May 2023**

This draft is issued to allow comments from interested parties; all comments will be given consideration prior to publication. No acknowledgement will normally be sent. **See overleaf for information on the submission of comments.**

No copying is allowed, in any form, without prior written permission from BSI except as permitted under the Copyright, Designs and Patent Act 1988 or for circulation within a nominating organization for briefing purposes. Electronic circulation is limited to dissemination by e-mail within such an organization by committee members.

Further copies of this draft may be purchased from BSI Shop <http://shop.bsigroup.com> or from BSI Customer Services, Tel: +44(0) 20 8996 9001 or email cservices@bsigroup.com. British, International and foreign standards are also available from BSI Customer Services.

Information on the co-operating organizations represented on the committees referenced above may be obtained from <http://standardsdevelopment.bsigroup.com>

Responsible Committee Secretary: Ms Charmaine Mukherjee (BSI)

Direct tel:

E-mail: charmaine.mukherjee@bsigroup.com

Introduction

This draft standard is based on national and international discussions. Your comments on this draft are invited and will assist in the preparation of the consequent standard.

For international standards, comments will be reviewed by the relevant UK national committee before sending the consensus UK vote and comments to the international committee, which will then decide appropriate action. If the international standard is approved, it is usual for the text to be published as a British Standard.

For national standards, comments will be reviewed by the relevant UK national committee and the resulting standards published as a British Standard.

UK Vote

Please indicate whether you consider the UK should submit a negative (with supporting technical reasons) or positive vote on this draft. Please indicate if you are aware of any reason why this draft standard should not be published as a British Standard.

Submission of Comments

- **Annotated drafts are not acceptable and will be rejected.**
- All comments should be submitted online at <https://standardsdevelopment.bsigroup.com/>. You will need to register in order to comment.

Template for comments and secretariat observations

Date: xx/xx/20xx	Document: ISO/DIS xxxx
------------------	------------------------

1	2	(3)	4	5	(6)	7
MB	Clause No./ Subclause No./Annex (e.g. 3.1)	Paragraph/Figure/ Table/Note	Type of comment	Commend (justification for change) by the MB	Proposed change by the MB	Secretariat observations on each comment submitted
	3.1	Definition 1	ed	Definition is ambiguous and needs clarifying.	Amend to read '...so that the mains connector to which no connection...'	
	6.4	Paragraph 2	te	The use of the UV photometer as an alternative cannot be supported as serious problems have been encountered in its use in the UK.	Delete reference to UV photometer.	



65/992/CD

COMMITTEE DRAFT (CD)

PROJECT NUMBER:

IEC 63278-2 ED1

DATE OF CIRCULATION:

2023-03-10

CLOSING DATE FOR COMMENTS:

2023-06-02

SUPERSEDES DOCUMENTS:

65/915/NP, 65/937A/RVN

IEC TC 65 : INDUSTRIAL-PROCESS MEASUREMENT, CONTROL AND AUTOMATION

SECRETARIAT:

France

SECRETARY:

Mr Didier GIARRATANO

OF INTEREST TO THE FOLLOWING COMMITTEES:

SC 3D,SyC SM,ISO/IEC JTC 1/SC 41

PROPOSED HORIZONTAL STANDARD:



Other TC/SCs are requested to indicate their interest, if any, in this CD to the secretary.

FUNCTIONS CONCERNED:

☐ EMC

☐ ENVIRONMENT

☐ QUALITY ASSURANCE

☐ SAFETY

This document is still under study and subject to change. It should not be used for reference purposes.

Recipients of this document are invited to submit, with their comments, notification of

- any relevant patent rights of which they are aware and to provide supporting documentation,
- any local regulations or technical reasons that may exist and should be considered should this proposal proceed, recognizing that failure to address such requirements could result in the need for new or existing "in some countries" clauses. See AC/22/2007.

TITLE:

Asset Administration Shell for Industrial Applications – Part 2: Information meta model

NOTE FROM TC/SC OFFICERS:

Copyright © 2023 International Electrotechnical Commission, IEC. All rights reserved. It is permitted to download this electronic file, to make a copy and to print out the content for the sole purpose of preparing National Committee positions. You may not copy or "mirror" the file or printed version of the document, or any part of it, for any other purpose without permission in writing from IEC.

CONTENTS

FOREWORD.....	6
INTRODUCTION.....	8
0.1 General.....	8
0.2 Overview on parts of the series.....	8
0.3 Information meta models.....	9
1 Scope.....	11
2 Normative references	11
3 Terms, definitions, abbreviated terms, and conventions.....	12
3.1 Terms and definitions.....	12
3.2 Abbreviated terms.....	15
3.3 Conventions.....	16
4 General considerations.....	16
5 Meta model of the Asset Administration Shell	17
5.1 Logical dependencies between part 1 and part 2 of the IEC 63278 series	17
5.2 Overview UML model.....	18
5.3 Detailed entities of the information meta model.....	20
5.3.1 General	20
5.3.2 Common Attributes	20
5.3.3 AssetAdministrationShell attributes.....	31
5.3.4 Asset Information Attributes.....	32
5.3.5 Submodel attributes.....	34
5.3.6 SubmodelElement attributes	35
5.3.7 Overview of Submodel Element Types	36
5.3.8 Environment	51
5.3.9 Referencing in Asset Administration Shells.....	51
5.3.10 Primitive and Simple Data Types	62
5.3.11 Cross Constraints and Invariants	72
5.4 Data Specification Templates.....	73
5.4.1 Introduction	73
5.4.2 Data Specification Template Attributes	75
Annex A UML style (informative).....	76
A.1 OMG UML General	76
A.2 Notes to graphical representation	80
Annex B Style of Detailed entities (informative).....	84
Annex C Traceability (mapping) to IEC 63278-1 requirements	85
Bibliography.....	90

FIGURES

41		
42		
43	Figure 1 – Relationships between the documents	8
44	Figure 2 – Example of four-layer meta model hierarchy described in [ISO/IEC 19505-	
45	1:2012]	10
46	Figure 3 – Detailed overview of Asset Administration Shell and related roles in IEC	
47	63278-1 clause 4 (IEC 63278-1 Figure 4).....	17
48	Figure 4 – Overview of information meta model of Asset Administration Shell	18
49	Figure 4 Overview of information model of Asset Administration Shell with inherited	
50	attributes	19
51	Figure 6 Meta model of AdministrativeInformation	20
52	Figure 7 Meta model of HasDataSpecification	22
53	Figure 7 Meta model of HasExtensions	23
54	Figure 9 Meta model of HasKind	24
55	Figure 10 Meta model of semantic references (HasSemantics)	25
56	Figure 11 Meta model of Identifiable	26
57	Figure 12 Meta model of Qualifiable	27
58	Figure 13 Meta model of Qualifiers	27
59	Figure 14 Meta model of Referable	29
60	Figure 15 Meta model AssetAdministrationShell	31
61	Figure 16 Meta model of Asset Information	32
62	Figure 17 Meta model of Submodel.....	34
63	Figure 18 Meta model of SubmodelElement	35
64	Figure 19 Meta model overview for SubmodelElement subtypes	36
65	Figure 20 Meta model of AnnotatedRelationshipElement.....	36
66	Figure 20 Meta model of BasicEventElement	37
67	Figure 22 Meta model of Blob	40
68	Figure 23 Meta model of Capability	41
69	Figure 24 Meta model of DataElement	41
70	Figure 25 Meta model of Entity	42
71	Figure 26 Meta model of Events.....	43
72	Figure 27 Meta model of File	44
73	Figure 28 Meta model of MultiLanguageProperty	44
74	Figure 29 Meta model of Operation	45
75	Figure 30 Meta model of Property	46
76	Figure 31 Meta model of ReferenceElement	46
77	Figure 32 Meta model of RelationshipElement	47
78	Figure 33 Meta model of SubmodelElementCollection	48
79	Figure 34 Meta model of SubmodelElementList	49
80	Figure 36 Meta model for Environment	51
81	Figure 36 Meta model of Reference	52
82	Figure 38 Meta model of Keys	53
83	Figure 38 Logical Model for Keys of References (non normative)	54

84	Figure 39 Meta model of KeyTypes enumeration.....	54
85	Figure 41 DataTypeDefXsd Enumeration	66
86	Figure 42 DefTypeDefRdf Enumeration.....	66
87	Figure 42 Built-In Types of XML Schema Definition 1.1 (XSD)	71
88	Figure 44 Core Elements of Using Data Specifications (non-normative)	74
89	Figure 44 Data Specification Templates	75
90	Figure 45 Class	76
91	Figure 46 Inheritance/Generalization	76
92	Figure 47 Multiplicity	77
93	Figure 48 Ordered Multiplicity	77
94	Figure 49 Association	77
95	Figure 50 Composition (composite aggregation)	78
96	Figure 51 Aggregation	78
97	Figure 52 Navigable Attribute Notation for Associations	78
98	Figure 53 Dependency	78
99	Figure 54 Abstract Class	79
100	Figure 55 Package	79
101	Figure 56 Imported Package	79
102	Figure 57 Enumeration	79
103	Figure 58 Data Type	80
104	Figure 59 Primitive Data Type	80
105	Figure 60 Note	80
106	Figure 61 Constraint	80
107	Figure 62 Graphical Representations of Composite Aggregation/Composition	81
108	Figure 63 Graphical Representation of Shared Aggregation	82
109	Figure 64 Graphical Representation of Generalization/Inheritance	82
110	Figure 65 Graphical Representation for Enumeration with Inheritance	83
111		
112		

TABLES

113	
114	Table 1 – Access modes for standardized information according to use-cases and
115	requirements of the stakeholders 16
116	Table 2 – Entities in conceptual framework to be scope in AAS meta model 17
117	Table 3 Simple data types used in meta model 62
118	Table 4 Primitive data types used in meta model 63
119	Table 5 Data Types with Examples 66
120	Table 6 – Template for document classes (elements)..... 84
121	Table 7 – Template for enumerations 84
122	Table 8 – Traceability to IEC 63278-1 requirements..... 85
123	
124	

INTERNATIONAL ELECTROTECHNICAL COMMISSION

ASSET ADMINISTRATION SHELL FOR INDUSTRIAL APPLICATIONS**Part 2: Information meta model****FOREWORD**

- 1) The International Electrotechnical Commission (IEC) is a worldwide organization for standardization comprising all national electrotechnical committees (IEC National Committees). The object of IEC is to promote international co-operation on all questions concerning standardization in the electrical and electronic fields. To this end and in addition to other activities, IEC publishes International Standards, Technical Specifications, Technical Reports, Publicly Available Specifications (PAS) and Guides (hereafter referred to as "IEC Publication(s)"). Their preparation is entrusted to technical committees; any IEC National Committee interested in the subject dealt with may participate in this preparatory work. International, governmental and non-governmental organizations liaising with the IEC also participate in this preparation. IEC collaborates closely with the International Organization for Standardization (ISO) in accordance with conditions determined by agreement between the two organizations.
- 2) The formal decisions or agreements of IEC on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested IEC National Committees.
- 3) IEC Publications have the form of recommendations for international use and are accepted by IEC National Committees in that sense. While all reasonable efforts are made to ensure that the technical content of IEC Publications is accurate, IEC cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.
- 4) In order to promote international uniformity, IEC National Committees undertake to apply IEC Publications transparently to the maximum extent possible in their national and regional publications. Any divergence between any IEC Publication and the corresponding national or regional publication shall be clearly indicated in the latter.
- 5) IEC itself does not provide any attestation of conformity. Independent certification bodies provide conformity assessment services and, in some areas, access to IEC marks of conformity. IEC is not responsible for any services carried out by independent certification bodies.
- 6) All users should ensure that they have the latest edition of this publication.
- 7) No liability shall attach to IEC or its directors, employees, servants or agents including individual experts and members of its technical committees and IEC National Committees for any personal injury, property damage or other damage of any nature whatsoever, whether direct or indirect, or for costs (including legal fees) and expenses arising out of the publication, use of, or reliance upon, this IEC Publication or any other IEC Publications.
- 8) Attention is drawn to the Normative references cited in this publication. Use of the referenced publications is indispensable for the correct application of this publication.
- 9) Attention is drawn to the possibility that some of the elements of this IEC Publication may be the subject of patent rights. IEC shall not be held responsible for identifying any or all such patent rights.

IEC 63278-2 has been prepared by IEC technical committee 65: INDUSTRIAL-PROCESS MEASUREMENT, CONTROL AND AUTOMATION. It is an International Standard.

The text of this International Standard is based on the following documents:

Draft	Report on voting
XX/XX/FDIS	XX/XX/RVD

Full information on the voting for its approval can be found in the report on voting indicated in the above table.

The language used for the development of this International Standard is English

This document was drafted in accordance with ISO/IEC Directives, Part 2, and developed in accordance with ISO/IEC Directives, Part 1 and ISO/IEC Directives, IEC Supplement, available

175 at https://www.iec.ch/members_experts/refdocs. The main document types developed by IEC
176 are described in greater detail at <https://www.iec.ch/standardsdev/publications>.

177 The committee has decided that the contents of this document will remain unchanged until the
178 stability date indicated on the IEC website under [webstore.iec.ch](https://www.iec.ch/webstore) in the data related to the
179 specific document. At this date, the document will be

- 180 • reconfirmed,
- 181 • withdrawn,
- 182 • replaced by a revised edition, or
- 183 • amended.

184

207 ○ Part 2 formalizes the requirements of Part 1 for Submodels and Submodel templates
208 to a clearly defined information metamodel. Furthermore, Part 2 considers the
209 specific requirements provided by Part 3.

210 ○ Part 3 addresses the cross-cutting concern security. Part 3 formalizes the
211 requirements of Part 1 and use cases of Part 4 and considers the modelling concepts
212 provided by Part 2 and functions provided by a further Part “interface”.

213 NOTE Further cross-cutting concerns could be considered in the future, such as uniform version management.

214 ○ Part 5 will formalize the requirements of Part 1 for the AAS interface to a clearly
215 defined set of functions. Part 5 is also subject to guidance of Part 2 and considers
216 the specific requirements provided by Part 3.

217 • The implementation is to be distinguished from the concept and design. Content that affects
218 the implementation is, for example, a REST implementation of the AAS interface or the
219 serialization of information represented by an AAS in an aasx-file. Currently the
220 implementation view is not addressed by the IEC 63278 series.

221

222 **0.3 Information meta models**

223 Information meta models are thought to be models of information models. Information meta
224 models are expected:

- 225 – to specify, how information is expressed in information models and how information is
226 accessed, if multiple software applications access standardized information,
- 227 – to enable syntactic interoperability [ISO/IEC 21823-1],
- 228 – to describe a user model.

229 Recognized layering of meta models is expressed in [ISO/IEC 19505-1:2012], developing a
230 four-layer meta model hierarchy, see Figure 1.

231

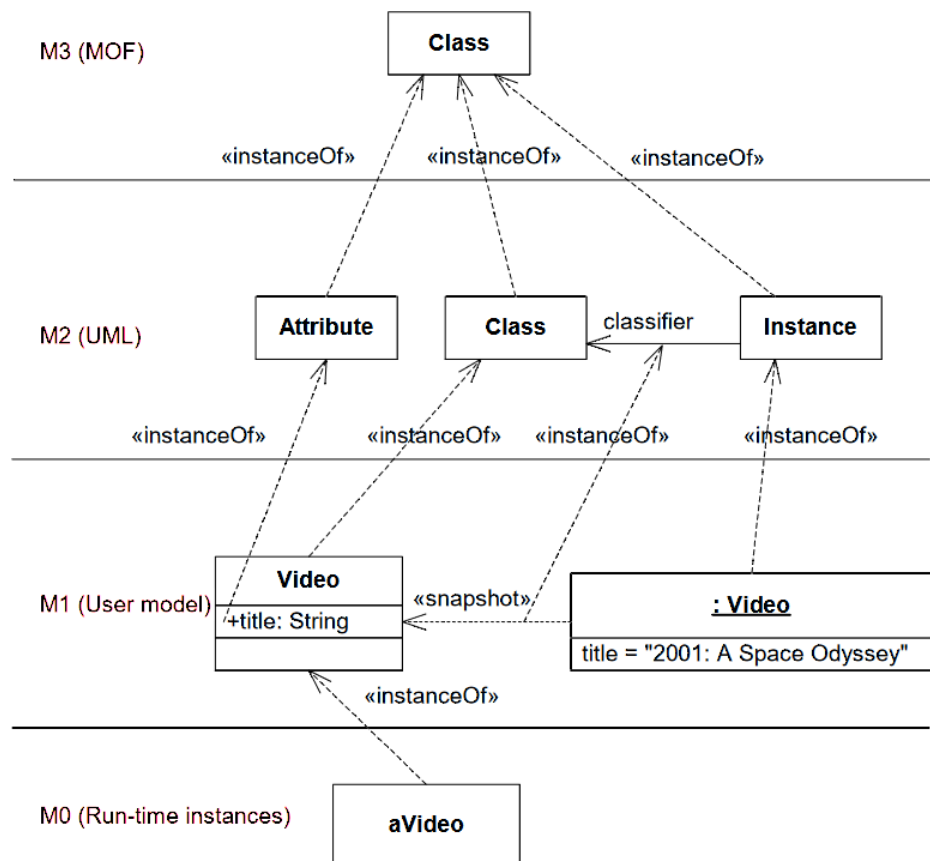


Figure 2 – Example of four-layer meta model hierarchy described in [ISO/IEC 19505-1:2012]

UML meta model has traditionally been based on such hierarchy, in [ISO/IEC 19505-1:2012] each layer is described the following:

- **M3 Meta Object Facility (MOF)** is referred as a meta-meta model, as it defines the language for specifying a meta model. It provides a metadata management framework and associated set of metadata services. It is though strictly speaking a meta model.
- **M2 Unified Modeling Language (UML)** corresponds to meta model layer. Its abstract syntax is specified by upper layer M3 and thus is referred as one instance of MOF meta-meta model. Its typical role is to define semantics of model elements instantiation in a model.
- **M1 User model** refers to instance of meta model. It defines languages describing semantic domains.
- **M0** bottom layer corresponds to Run-time instance of modelled elements from M1 layer.

ASSET ADMINISTRATION SHELL FOR INDUSTRIAL APPLICATIONS

Part 2: Information meta model

1 Scope

This part of IEC 63278 specifies the meta model of Asset Administration Shell (AAS). It specifies meta model entities defining how

- Submodel(s) are listed in an AAS;
- Submodel(s) resp. Submodel template(s) are structured;
- the purpose and the specific content of SubmodelElements resp. Submodel template elements is defined. The meta model therefore provides the basis for specifying Submodel templates, which guide the creation of Submodels;
- references to concept repository entries of concept repositories are defined;
- the AAS management information is defined; this includes how the association of an AAS to the asset is defined.

This document is the basis for Part 5 defining the functions of the AAS interface.

This document does not describe implementation aspects, see Figure 1.

NOTE This document will allow the creation of data mappings to suitable technologies to be used in the implementation view for the different life cycle phases of an asset such as: XML, JSON, RDF, AutomationML (IEC 62714) and OPC UA (IEC 62541).

2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

IEC TR 63283-1 ED1: Industrial-process measurement, control and automation – Smart Manufacturing – Part 1: Terms and definitions

IEC 63278-1: Asset Administration Shell for industrial applications – Part 1: Asset Administration Shell structure

IETF RFC 2046: Multipurpose Internet Mail Extensions (MIME) – Part Two: Media Types

IETF RFC 8089: The "file" URI Scheme

ISO/IEC 19505-2:2012, *Information technology — Object Management Group Unified Modeling Language (OMG UML) — Part 2: Superstructure*

284 W3C: XML Schema 1.1

285 **3 Terms, definitions, abbreviated terms, and conventions**

286 **3.1 Terms and definitions**

287 For the purposes of this document, the following terms and definitions apply.

288 ISO and IEC maintain terminological databases for use in standardization at the following addresses:

- 289 – IEC Electropedia: available at <http://www.electropedia.org/>
- 290 – ISO Online browsing platform: available at <http://www.iso.org/obp>

291 ***The following definitions include a selected set of the terms and definition as defined in IEC***
292 ***63278-1.***
293

294 **3.1.1**

295 **asset**

296 physical, digital, or intangible entity that has value to an individual or an organization

297 [SOURCE: IEV 741-01-04 (modified)]

298 @Editors: please review latest version of the IEC 63283-1 for the latest definition.

299 Note 1 to entry: An asset can be single entity, a collection of entities, an assembly of entities or a composition of
300 entities.

301 EXAMPLE 1 Examples for physical entities are equipment, raw material, parts components and pieces, supplies,
302 consumables, physical products and waste.

303 EXAMPLE 2 Software is an example of a digital asset.

304 EXAMPLE 3 A software license is an example of an intangible asset.

305 **3.1.2**

306 **Asset Administration Shell**

307 **AAS**

308 standardized digital representation of an asset

309 **3.1.3**

310 **AAS interface**

311 interface of an AAS giving uniform access to information and services

312 EXAMPLE Examples for services are exploration services, asset services, and asset related services.

313 **3.1.4**

314 **AAS responsible**

315 Individual or organization having interest in an asset and governing an Asset Administration Shell

316 **3.1.5**

317 **AAS user application**

318 software application which accesses an AAS via its AAS interface(s) for use by humans or for automatic
319 processing

320 **3.1.6**

321 **asset integration**

322 software and/or computing infrastructure needed to access asset services

3.1.7**asset service**

service that is provided by the considered asset

3.1.8**asset related service**

service that is not provided by the considered asset, but by software and/or by computing infrastructure outside of the considered asset

3.1.9**attribute**

data element for the computer-sensible description of a property, a relation or a class

[SOURCE: ISO/IEC Guide 77-2]

3.1.10**class**

description of a set of objects that share the same attributes, operations, methods, relationships, and semantics

[SOURCE: IEC TR 62390:2005-01, 3.1.4]

3.1.11**concept**

unit of knowledge created by a unique combination of characteristics

[SOURCE: IEC 61360-1:2016, 3.1.8] 3D/265/CDV

3.1.12**concept repository**

collection of entries that allows lookup by concept identifier and where relationships between entries can be described

[SOURCE: IEC 62832-1:2020, 3.1.5, modified – "concept dictionary" deleted]

3.1.13**concept repository entry**

description of a concept containing, at a minimum, an unambiguous concept identifier, a preferred name, and a description

[SOURCE: IEC 62832-1:2020, 3.1.6]

3.1.14**dereferencing**

act of retrieving a digital representation based on an identifier of the entity

3.1.15**digital representation**

information and services representing an entity from a given viewpoint

EXAMPLE 1 Examples of information are properties (e.g. maximum temperature), actual parameters (e.g. actual velocity), events (e.g. notification of status change), schematics (electrical) and visualization information (2D drawings, 3D drawing).

EXAMPLE 2 Examples of services are providing the history of the configuration data, providing the actual velocity, and providing a simulation.

EXAMPLE 3 Examples of viewpoints are mechanical, electrical, or commercial characteristics.

3.1.16**entity**

thing (physical or non-physical) having a distinct existence

[SOURCE: ISO/IEC FDIS 20924, 3.1.18]

3.1.17**instance asset**

specific asset that is uniquely identifiable

Examples of instance assets are material, a product, a part, a device, a machine, software, a control system, or a production system.

3.1.18**interface**

shared boundary between two entities defined by functional characteristics, signal characteristics, or other characteristics as appropriate

3.1.19**interoperability**

capability of two or more entities to exchange items in accordance with a set of rules and mechanisms implemented by an interface in each entity, in order to perform their respective tasks

Note 1 to entry: Examples of entities include devices, equipment, machines, people, processes, applications, software units, systems and enterprises.

Note 2 to entry: Examples of items include information, material, energy, control, assets and ideas.

3.1.20**meta model**

special kind of model that specifies the abstract syntax of a modelling language

[SOURCE: IEC 19506:2012, modified: Note 2 removed]

Note 1 to entry: A model is an instance of a meta model

3.1.21 property instance

information consisting at least of the identifier of a property type and a property value

Note 1 to entry: The concept of type and instance applies to properties. If omitted, the term property refers to property types.

Note 2 to entry: The property instances have a value which might be provided by the manufacturer or another partner in the value chain. Sometimes, a property instance exists without a specific value, e.g. giving an existential statement.

Note 3 to entry: A property instance is also called property-value pair in certain standards. A property instance is also called data element in some standards.

3.1.22**property type**

defined parameter suitable for the description and differentiation of products

[SOURCE: ISO/IEC Guide 77-2, modified: (type) was added to the term]

Note 1 to entry: The concept of type and instance applies to properties. If omitted, the term property refers to property types.

Note 2 to entry: The property types are defined in dictionaries (like IEC Common Data Dictionary or ECLASS), they do not have a value. The property type is also called data element type in some standards.

3.1.23**role**

set of characteristics that distinguish an entity's ability to exhibit a set of required behaviors

[SOURCE: ISO 18435-1:2009, 3.22]

3.1.24**service**

distinct part of the functionality that is provided by an entity through interfaces

[SOURCE: IEV 741-01-28]

3.1.25**software application**

software functional element specific to the solution of a problem in industrial-process measurement and control

[SOURCE: IEC TR 62390:2005-01, 3.1.2; modified: software was added to the term]

Note 1 to entry: A software application may be distributed among resources and may communicate with other software applications.

3.1.26**Submodel**

representation of an aspect of an asset

3.1.27**SubmodelElement**

element of a Submodel

3.1.28**Submodel template**

template for the representation of an aspect of an asset

3.1.29**Submodel template element**

element of a Submodel template

3.1.30**type asset**

(abstract) representation of a set of instance assets with common characteristics and features

Note 1 to entry: The set of instance assets may exist or may not exist.

Examples of type assets are type of material, a product type, a type of a part, a device type, a machine type, a type of software, a type of control system, a type of production system.

3.2 Abbreviated terms

Table from IEC 63278-1 with proposal which to remove because not used in this part.

AAS	Asset Administration Shell
API	Application Programming Interface
AML	AutomationML, Automation Markup Language
DF	Digital Factory
GUID	Globally Unique Identifier
ICT	Information and Communications Technology
IEC	International Electrotechnical Commission

ID	Identifier
IEC CDD	IEC Common Data Dictionary (IEC 61360)
IRDI	International registration data identifier
ISO	International Organization for Standardization
IT	Information technology
UML	Unified Modeling Language
URI	Uniform Resource Identifier
XML	Extensible Markup Language

442

443 **3.3 Conventions**

444 For some terms, such as Asset Administration Shell, a special notation is used, in order to
 445 distinguish them from English text.

446 In Subsection 5.3 a tabular text format is used to specify the meta model of elements further.
 447 These tabular texts are considered inline text and not designated a table number.

448 **4 General considerations**

449 According to the specific use-cases and requirements of the stakeholders, the standardized
 450 information access with respect to Asset Administration Shells can be fulfilled by different
 451 access modes (see Table 1).

452 **Table 1 – Access modes for standardized information according to use-cases and**
 453 **requirements of the stakeholders**

Access mode	Quality of access to standardized information	Example
Type 1	Information can be written in a standardized way; another software application is able to read and process the information accordingly.	file
Type 2	A software application can query an external resource/ server and retrieve an expected set of information or trigger an intended operation.	client/ server interface
Type 3	A software application can interact with other external software applications and can achieve complex goals.	complex interaction between software application interfaces

454 NOTE According to the scope, no further definitions to technical implementations are given in this document. Type
 455 2 and 3 are expected to be subject of further parts of the IEC 63278 series.

5 Meta model of the Asset Administration Shell

5.1 Logical dependencies between part 1 and part 2 of the IEC 63278 series

The conceptual framework of the AAS in IEC 63278-1 gives a detailed overview of Asset Administration Shell and related roles (see Figure 3). From this conceptual framework, entities described in Table 2 are selected to be further specified in the meta model.

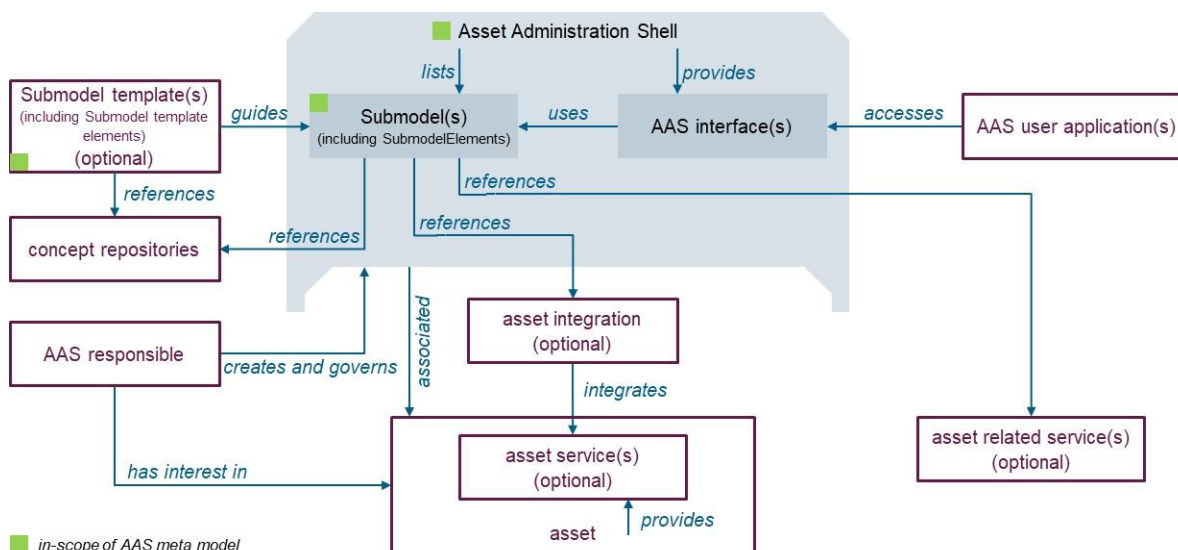


Figure 3 – Detailed overview of Asset Administration Shell and related roles in IEC 63278-1 clause 4 (IEC 63278-1 Figure 4)

NOTE 1 AAS management information is not shown in Figure 4 of Part 1.

NOTE 2 The guidance of a Submodel by a Submodel template is addressed in Part 4.

Table 2 – Entities in conceptual framework to be scope in AAS meta model

Entity in conceptual framework	scope AAS meta model	Further remarks
Asset Administration Shell	yes	
Submodel	yes	
SubmodelElements	yes	
references between entities	yes	references expected to be suitable for entities in concept repositories, asset integration, asset related services
asset	no	
AAS responsible	no	identifiers of AAS responsible in scope
asset integration, asset service, asset related services	no	subject of references
Submodel templates, Submodel template elements	yes	
AAS user application	no	

5.2 Overview UML model

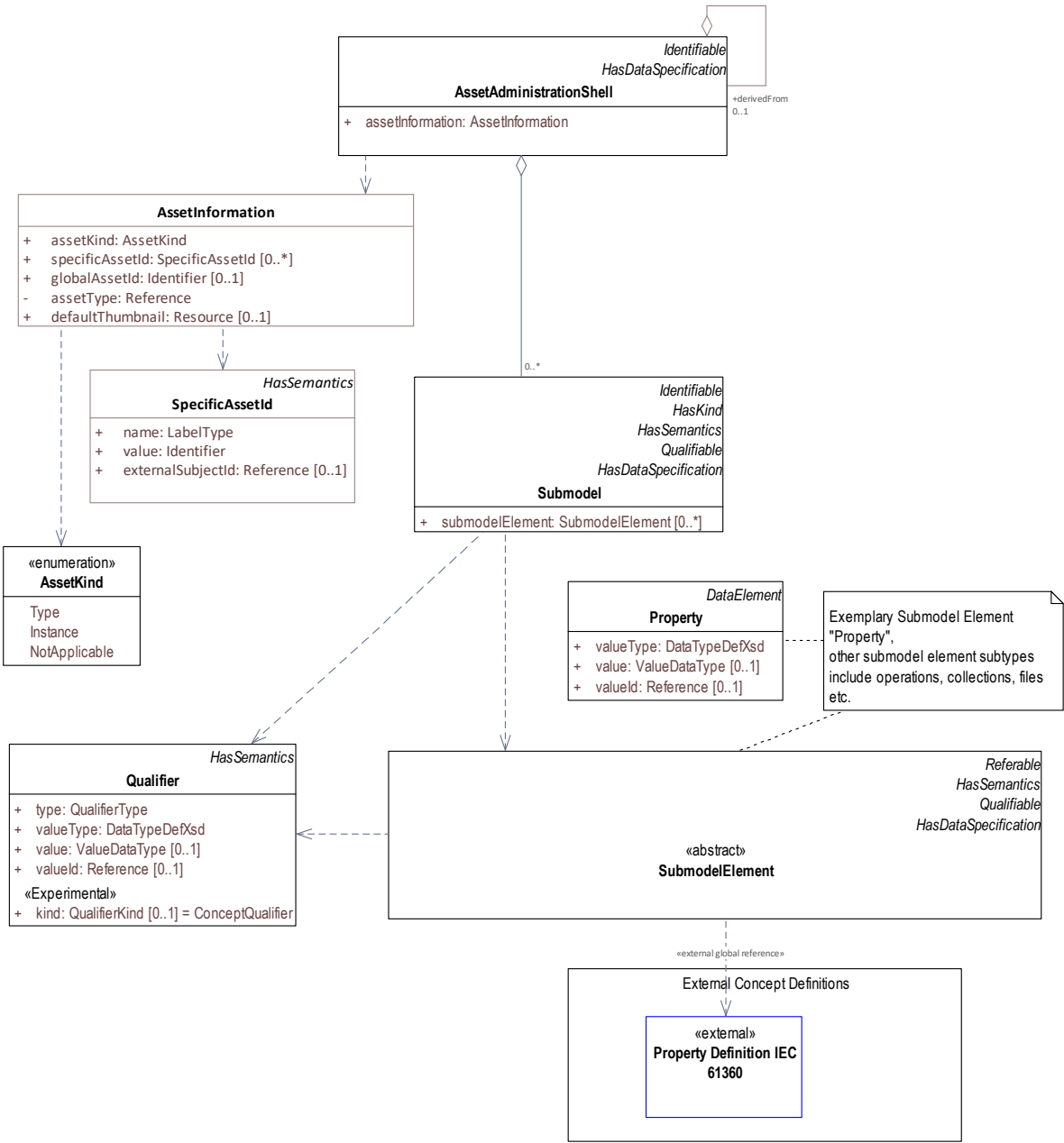


Figure 4 – Overview of information meta model of Asset Administration Shell

[illegible]

In Figure 4 an overview of the information meta model of the Asset Administration Shell with its most important classes is given. Details on the graphical UML representation of the meta model can be found in Annex A.

- The Asset Administration Shell owns information about the asset it is representing (AssetInformation).
- The Asset Administration Shell references zero or more Submodels.
- A Submodel is composed of zero or many SubmodelElements.

The other classes and the attributes are defined in detail in the following subsection.

– **Identifiables** are elements within the information model that have the attribute “identification”. Only Identifiables have administrative information (AdministrativeInformation).

– **Referables** are elements within the information model that have the attribute “idShort”.

There are two elements within the information model having the attribute “semanticId”: the Submodel and the particular SubmodelElements. Elements with this attribute are elements for which semantics can be assigned to by referencing an entry in a corresponding concept repository.

5.3 Detailed entities of the information meta model

5.3.1 General

In this clause the classes of the meta model are specified in detail. For an overview see Clause 5.2. In Annex A, UML modelling is explained together with the specifics used in this specification. In Annex B, the templates used to describe the classes and relationships are explained. In Clause 5.2, Figure 4, particular elements are shown together with all inherited attributes to give a complete overview.

For understanding the specifications, it is crucial to understand the common attributes first (Clause 5.3.2). They are reused throughout the specifications of the other classes (“inherits from”) and define important concepts like identifiable, qualifiable etc. They are abstract, i.e. there is no object instance of such classes.

Another important concept is the concept of referencing and how a reference is represented in the UML diagrams and the tables. This is explained in Clause 5.3.9.

Constraints that are no invariants of classes are specified in Clause 5.3.11.

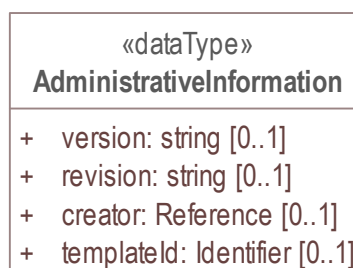
5.3.2 Common Attributes

5.3.2.1 General

In the clause the abstract classes that represent commonly used attributes and terminology are specified together with the classes and data types exclusively used in these classes. The order they are represented is alphabetically.

5.3.2.2 AdministrativeInformation attributes

Figure 6 Meta model of AdministrativeInformation



Every *Identifiable* may have administrative information. Administrative information includes for example

- Information about the version of the element
- Information about who created or who made the last change to the element

- Information about the languages available in case the element contains text, for translating purposed also the master or default language may be defined

In the current version of the Asset Administration Shell meta model only version information is defined for administrative information. In later versions additional attributes may be added.

Version corresponds in principle to the *version_identifier* according to IEC 62832 but is not used for concept identifiers only (IEC TS 62832-1) but for all identifiable elements. Version and revision together correspond to the version number according to IEC 62832.

For other attributes of the administrative information like creator we refer to ISO 15836-1:2017, the Dublin Core metadata element set.

AdministrativeInformation allows the usage of templates (*HasDataSpecification*). Data Specifications are defined in separate documents.

NOTE 1 Two Submodels with the same semanticId but different AdministrativeInformation (this especially includes different versions), shall have different IDs (Submodel/id). The idShort or numeric identifier typically do not change, i.e. they are identical. The same holds for other Identifiables (Identifiable/id).

NOTE 2 Since Submodels with different version shall have different identifiers it is possible that an Asset Administration Shell has two Submodels with the same semanticId but different versions, i.e. different Administrative Information.

NOTE 3 Some of the AdministrativeInformation like the version string might need to be part of the identification. This is similar to the handling of identifiers of concept repository entries using IRDIs. In ECLASS the IRDI 0173-1#02-AO677#002 contains the version information #002.

NOTE 4 The process of versioning of or adding other administrative information to elements is done by external version or configuration management software and not by the Asset Administration Shell itself.

Class:	AdministrativeInformation <<DataType>>		
Explanation:	Administrative meta information for an element like version information. <u>Constraint AASd-005:</u> If <i>AdministrativeInformation/version</i> is not specified than also <i>AdministrativeInformation/revision</i> shall be unspecified. This means, a revision requires a version. If there is no version there is no revision neither. Revision is optional. <u>Constraint AASd-135:</u> <i>AdministrativeInformation/version</i> shall have a length of maximum 4 characters. <u>Constraint AASd-136:</u> <i>AdministrativeInformation/revision</i> shall have a length of maximum 4 characters.		
Inherits from:	HasDataSpecification		
Attribute	Explanation	Type	Cardinality
version	Version of the element.	string	0..1
revision	Revision of the element.	string	0..1
creator	The subject ID of the subject responsible for making the element.	Reference	0..1
templateId	Identifier of the template that guided the creation of the element. NOTE 1 In case of a Submodel the templateId is the identifier of the Submodel template that guided the creation of the Submodel.	Identifier	0..1

Class:	AdministrativeInformation <<DataType>>		
	NOTE 2 The Submodel templateId is not relevant for validation. For validation the <i>Submodel/semanticId</i> shall be used.		
	NOTE 3 Usage of templateId is not restricted to Submodel instances. The creation of Submodel templates can be guided by another Submodel template, as well.		

544

545 5.3.2.3 HasDataSpecification attributes

546 Figure 7 Meta model of HasDataSpecification



547

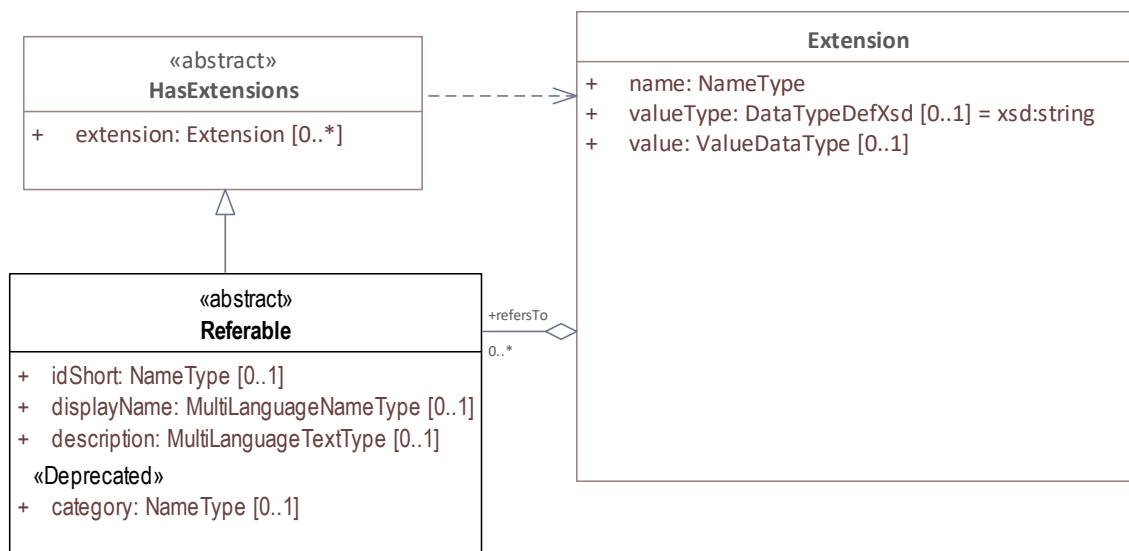
Class:	HasDataSpecification <<abstract>>		
Explanation:	Element that can be extended by using data specification templates. A data specification template defines a named set of additional attributes an element may or shall have. The data specifications used are explicitly specified with their global ID.		
Inherits from:	--		
Attribute	Explanation	Type	Cardinality
dataSpecification	Global reference to the data specification template used by the element.	Reference	0..*

548

549 For more details on data specifications see Clause 5.4.

5.3.2.4 HasExtensions attributes

Figure 8 Meta model of HasExtensions



See Clause 5.3.11.4 for constraints on extensions.

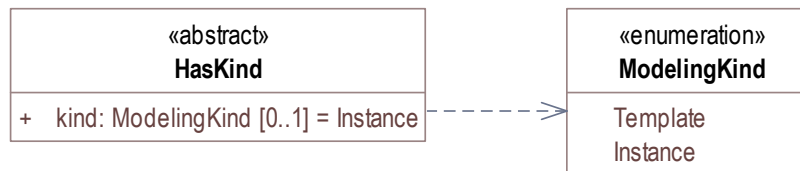
Class:	HasExtensions <<abstract>>		
Explanation:	Element that can be extended by proprietary extensions. NOTE Extensions are proprietary, i.e. they do not support global interoperability.		
Inherits from:	-		
Attribute	Explanation	Type	Cardinality
extension	An extension of the element.	Extension	0..*

Class:	Extension		
Explanation:	Single extension of an element.		
Inherits from:	HasSemantics		
Attribute	Explanation	Type	Cardinality
name	Name of the extension.	NameType	1
valueType	Type of the value of the extension. Default: xs:string	DataTypeDefXsd	0..1
value	Value of the extension	ValueDataType	0..1

Class:	Extension		
refersTo	Reference to an element the extension refers to.	ModelReference<Referable>	0..*

5.3.2.5 HasKind attributes

Figure 9 Meta model of HasKind



Class:	HasKind		
Explanation:	An element with a kind is an element that can either represent a template or an instance. Default for an element is that it is representing an instance.		
Inherits from:	--		
Attribute	Explanation	Type	Cardinality
kind	Kind of the element: either type or instance. Default value = <i>Instance</i>	ModellingKind	0..1

The kind enumeration is used to denote whether an element is of kind Template or Instance.

Enumeration:	ModellingKind
Explanation:	Enumeration for denoting whether an element is a template or an instance.
Set of:	--
Literal	Explanation
Template	Alternative 1 Software element which specifies the common attributes shared by all instances of the template. [SOURCE: IEC TR 62390:2005-01, 3.1.25] modified Alternative 2 Element is a Submodel template or Submodel template element.

Enumeration:	ModellingKind
Instance	Alternative 1 Concrete, clearly identifiable component of a certain template. Note: It becomes an individual entity of a template, for example a device model, by defining specific property values. Note: In an object-oriented view, an instance denotes an object of a template (class). [SOURCE: IEC 62890:2016, 3.1.16 65/617/CDV] modified Alternative 2 Element is not a Submodel template or Submodel template element.

5.3.2.6 HasSemantics attributes

Figure 10 Meta model of semantic references (HasSemantics)

«abstract» HasSemantics
+ semanticId: Reference [0..1] + supplementalSemanticId: Reference [0..*]

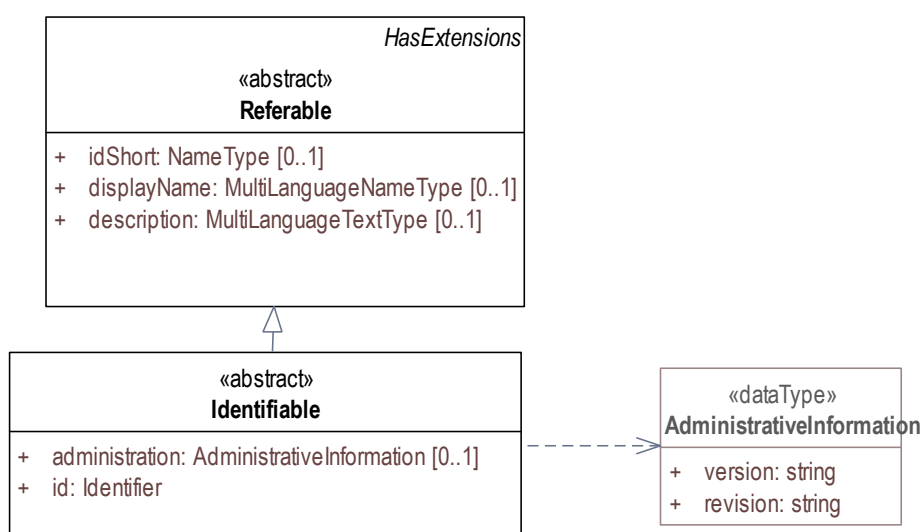
Class:	HasSemantics <<abstract>>		
Explanation:	Element that can refer to a concept repository entry plus some supplemental concept repository entries. <u>Constraint AASd-118</u>: If there is a supplemental semantic ID (<i>HasSemantics/supplementalSemanticId</i>) defined then there shall be also a main semantic ID (<i>HasSemantics/semanticId</i>).		
Inherits from:	--		
Attribute	Explanation	Type	Cardinality
semanticId	Identifier of the semantic definition of the element, that is, the concept repository entry. It is called semantic ID or also main semantic ID of the element. It is recommended to use a global reference.	Reference	0..1
supplementalSemanticId	Identifier of a supplemental semantic definition of the element, that is, a further concept repository entry. It is called	Reference	0..*

Class:	HasSemantics <<abstract>>		
	supplemental semantic ID of the element.		
	It is recommended to use a global reference.		

565

566 5.3.2.7 Identifiable attributes

567 Figure 11 Meta model of Identifiable



568 An Identifiable is a Referable with a globally unique identifier (*Identifier*). To reference an
 569 Identifiable only the global ID (*Identifiable/id*) shall be used because the *idShort* is not unique
 570 for an Identifiable. Identifiables may have *AdministrativeInformation* such as version.

571 Referable elements not being Identifiable can be referenced, but for doing so the context of the
 572 element is needed. A Referable not being Identifiable has a short identifier (*idShort*) that is
 573 unique just in its context, its name space.

574 NOTE 1 Confer Clause 5.3.11.2 for constraints on Referables and Identifiables.

575

Class:	Identifiable <<abstract>>		
Explanation:	An element that has a globally unique identifier.		
Inherits from:	Referable		
Attribute	Explanation	Type	Cardinality
administration	AdministrativeInformation of an identifiable element. NOTE Some of the administrative information like the version number might need to be part of the identification.	AdministrativeInformation	0..1