



DPC: 25/30483124 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: 25 September 2025

Origin: International

Latest date for receipt of comments: 17 November 2025

Project No. 2023/02736

Responsible committee: AMT/4 Industrial data and manufacturing interfaces

Interested committees:

Title: BS ISO 23726-3 Draft Automation systems and integration. Ontology based interoperability. Part 3: Industrial data ontology.

Please notify admin.start@bsigroup.com if you are aware of any keywords that might assist in clarifying or identifying the standard or of 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 17 November 2025

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 <https://standardsdevelopment.bsigroup.com/>.

E-mail: admin.start@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	Comment (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.	



DRAFT International Standard

ISO/DIS 23726-3

Automation systems and integration — Ontology based interoperability —

Part 3: Industrial data ontology

ICS: 35.240.50

ISO/TC 184/SC 4

Secretariat: **ANSI**

Voting begins on:
2025-09-24

Voting terminates on:
2025-12-17

This document has not been edited by the ISO Central Secretariat.

THIS DOCUMENT IS A DRAFT CIRCULATED FOR COMMENTS AND APPROVAL. IT IS THEREFORE SUBJECT TO CHANGE AND MAY NOT BE REFERRED TO AS AN INTERNATIONAL STANDARD UNTIL PUBLISHED AS SUCH.

IN ADDITION TO THEIR EVALUATION AS BEING ACCEPTABLE FOR INDUSTRIAL, TECHNOLOGICAL, COMMERCIAL AND USER PURPOSES, DRAFT INTERNATIONAL STANDARDS MAY ON OCCASION HAVE TO BE CONSIDERED IN THE LIGHT OF THEIR POTENTIAL TO BECOME STANDARDS TO WHICH REFERENCE MAY BE MADE IN NATIONAL REGULATIONS.

RECIPIENTS OF THIS DRAFT ARE INVITED TO SUBMIT, WITH THEIR COMMENTS, NOTIFICATION OF ANY RELEVANT PATENT RIGHTS OF WHICH THEY ARE AWARE AND TO PROVIDE SUPPORTING DOCUMENTATION.

Reference number
ISO/DIS 23726-3:2025(en)

Full document details: 25/30483124 DC

© ISO 2025  **Documents**
Document Sharing Hub



COPYRIGHT PROTECTED DOCUMENT

© ISO 2025

All rights reserved. Unless otherwise specified, or required in the context of its implementation, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

Page

Foreword	iv
Introduction	v
1 Scope	1
2 Normative references	1
3 Terms and definitions	1
4 Abbreviated terms	7
5 IDO entities	8
5.1 General	8
5.2 Declarations	11
5.2.1 Prefixes	11
5.2.2 Ontology	11
5.3 Classes	12
5.3.1 Disjointness clause	12
5.3.2 The Object taxonomy	12
5.3.3 The Dependent taxonomy	27
5.3.4 The Temporal taxonomy	34
5.3.5 Specified versus Actual	44
5.4 Object properties (individual-to-individual relations)	45
5.4.1 General	45
5.4.2 Parthood	45
5.4.3 Possession of quality	59
5.4.4 Participation in activity	61
5.4.5 Possession of realizable potential	65
5.4.6 Realization of potential in activity	70
5.4.7 Activity profile of quality	72
5.4.8 Physical connection	73
5.4.9 Relative location	74
5.4.10 Relative occurrence	78
5.4.11 Occurrence at a site	82
5.4.12 Representation in information objects	83
5.4.13 Concretization of information	88
5.4.14 Implementation of specified as actual	90
5.5 Data properties (individual-to-value relations)	92
5.5.1 <code>lis:datumValue</code>	92
5.5.2 <code>lis:qualityQuantityValue</code>	92
5.6 Annotation properties (informative-only relations)	93
5.6.1 General	93
5.6.2 Annotation properties for naming RDF resources	93
5.6.3 Annotation properties for providing definitions	93
5.6.4 Annotation properties for informal guidelines	94
5.6.5 Annotation properties for cross-referencing other ontologies	94
5.6.6 Annotation properties for provenance and versioning	95
5.6.7 Annotation properties for representing ontology evolution	96
Annex A (informative) Modelling facts with IDO	97
Annex B (informative) Supplementary resources	103
Annex C (informative) Alignment of IDO with other ontologies	108
Bibliography	130

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of ISO documents should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

ISO draws attention to the possibility that the implementation of this document may involve the use of (a) patent(s). ISO takes no position concerning the evidence, validity or applicability of any claimed patent rights in respect thereof. As of the date of publication of this document, ISO had not received notice of (a) patent(s) which may be required to implement this document. However, implementers are cautioned that this may not represent the latest information, which may be obtained from the patent database available at www.iso.org/patents. ISO shall not be held responsible for identifying any or all such patent rights.

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see www.iso.org/iso/foreword.html.

This document was prepared by Technical Committee ISO/TC 184, *Automation systems and integration*, Subcommittee SC 4, *Industrial data*.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html.

Introduction

In the digital era, the complexity of industrial assets, such as a new offshore renewable energy fields, an existing sub-sea development, or an aircraft model, requires collaboration and interoperability across various industries and domains. In this context, many types of assets can be considered systems of systems, where each collaboration (e.g. development, operation, maintenance, and reconstruction of these assets) involves multiple stakeholders. Each stakeholder has specified the use of structured vocabularies (e.g. from ISO, IEC, ASME, DIN), data model(s) and design standards. The exchange of information at each stage of the life cycle, and within a life cycle stage, related to for example cost, schedule, design, operation, maintenance, and re-build, is therefore complex. Errors in information exchange can lead to risks, such as time delays, cost overruns, safety events and product delays. Management of these risks include controls such as facilitating automated data exchange. This semantic interoperability involves humans and computers being able to unambiguously understand data and its context and is achieved through the reasoning ability of an ontology.

Industrial Data Ontology (IDO) is an OWL 2 ontology. IDO is designed to support machine automated reasoning according to OWL 2 Direct Semantics over information used in the design and through-life operation of complex, long-life assets.

IDO provides 1. a language that is accessible to engineers and 2. industrially relevant modelling patterns, while retaining the technical capability to perform OWL DL reasoning with a well-founded ontological base. This combination of user-centric characteristics is the primary motivation for the standardisation of IDO. The taxonomy of classes in IDO is shown in [Figure 1](#).

IDO is suitable for industrial use cases, to create vocabularies and asset models and to exploit OWL DL reasoning for quality assurance and inference of implicit knowledge. Being anchored in W3C semantic technology standards, RDF graph assurance, such as (closed-world) SHACL validation, can be additionally used for enforcing context-specific structure and completeness of data conforming to IDO and its domain-specific extensions.

IDO provides a comprehensive set of fundamental elements to guide creation of ontology patterns and examples of patterns commonly used to capture important data and relations throughout the asset life cycle. The use, and reuse, of approved patterns reduces risks that modelling mistakes will be made and increases the speed of the model development and quality checking processes.

The intended users of IDO are software architects, data modellers, and data custodians involved in supporting engineers in the design, construction, manufacturing, operation, maintenance, and disposal of assets and discrete or continuous processing systems in the engineering sector. As semantic modelling is in the domain of computer science and software engineering, the content is aimed at those professionals.

IDO was developed in response to demand from the industrial engineering community for ontology-based solutions to the problem of semantic interoperability across information systems associated with the multiple stakeholders involved through the life cycle of assets.

To enable interoperability IDO offers two main root concepts. First a hierarchical classification system allows a structured organization of industrial data, facilitating comprehensive modelling and analysis. Second, a structured framework for modelling relationships between entities within specific contexts. This framework provides the language elements needed to enable domain experts to define a controlled and understandable data flow as well as enabling a comprehensible data analysis.

IDO has a track-record of use in the oil and gas sector. It has been used for OWL DL reasoning over material master data in the design phase as well as for representing as-built process plants with millions of parts. It enables annotation of work orders for use in language models. IDO is also coherent with, and supports, modelling done with ISO/IEC 81346-1:2022 which provides an important link for the users of these standards to the models on which the data they are referring to in ISO/IEC 81346 is based.

IDO draws on the heritage of existing ontologies published (or in the process of being published) by ISO/IEC such as ISO/IEC 21838 and ISO 15926.

The name and contact information of the maintenance agency for this document can be found at www.iso.org/maintenance_agencies.

- Object
 - FunctionalObject
 - System
 - InformationObject
 - QualityDatum
 - ScalarQuantityDatum
 - UnitOfMeasure
 - Scale
 - Location
 - Site
 - SpatialLocation
 - LineInSpace
 - PlaneInSpace
 - PointInSpace
 - VolumeInSpace
 - Organization
 - PhysicalObject
 - PhysicalArtefact
 - MateriallyClassifiedObject
 - Feature
 - InanimatePhysicalObject
 - Stream
 - Organism
 - Person
 - Phase
- Dependent
 - Potential
 - Disposition
 - Capability
 - Function
 - Interest
 - Role
 - Quality
 - MaterialCompositionQuality
 - PhaseQuality
 - PhysicalQuantity
 - ShapeQuality
- Temporal
 - Activity
 - ActivityProfile
 - Event
 - SpatiotemporalRegion
 - TemporalRegion
 - InstantRegion
 - Instant
 - IntervalRegion
 - Interval
- Actual
- Specified

Figure 1 — IDO class hierarchy

Automation systems and integration — Ontology based interoperability —

Part 3: Industrial data ontology

1 Scope

This document specifies the Industrial Data Ontology (IDO), a Web Ontology Language (OWL) ontology.

IDO is intended for industrial data and information, to build vocabularies and manage asset models which employ reference data libraries and exploit OWL DL. IDO is intended for use in all life cycle phases of industrial assets and processes.

The following are within the scope of this document:

- general overview of IDO and of the main principles;
- normative definitions of IDO classes and relations;
- informative examples of the use of IDO terminology in context;
- re-usable representation patterns.

The following are outside the scope of this document:

- specification of data modelling languages, such as RDF and OWL used in ontology development;
- specification of methods for reasoning with ontologies.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements 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.

W3C Recommendation — OWL 2 Web Ontology Language Direct Semantics (Second Edition)

W3C Recommendation — OWL 2 Web Ontology Language Structural Specification and Functional-Style Syntax (Second Edition)

3 Terms and definitions

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

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

- ISO Online browsing platform: available at <https://www.iso.org/obp>
- IEC Electropedia: available at <https://www.electropedia.org/>

3.1**annotation property**

RDF property (3.29) included in the vocabulary of an *OWL ontology* (3.21) with no semantic significance

Note 1 to entry: Annotation properties allow for including information in an ontology that is not subject to OWL reasoning.

3.2**asset model**

model (3.17) containing *individuals* (3.12), classified and related in accordance with a *reference data library* (3.30) and representing an industrial asset

Note 1 to entry: IDO is intended to be a suitable basic vocabulary for asset models represented as OWL 2 ontologies.

3.3**axiom**

statement that is taken to be true, to serve as a premise for further reasoning

Note 1 to entry: Axioms may be formulated as natural language sentences or as formulae in a formal language. In the OWL community, 'axiom' is used to refer to statements that say what is true in the domain that are 'basic' in the sense that they are not inferred from other statements.

[SOURCE: ISO 15926-2:2003, 3.9]

3.4**class**

general entity (3.10)

Note 1 to entry: In some ontology communities, all general entities are referred to as classes. In other ontology communities, a distinction is drawn between classes as the extensions of general entities (for example as sets of instances) and the general entities themselves, sometimes referred to as 'types', 'kinds', or 'universals'. The expression 'class or type' is used in what follows in order to remain neutral as between these different usages.

[SOURCE: ISO/IEC 21838-1:2019, 3.2]

3.5**data**

representation of *information* (3.13) in a formal manner suitable for communication, interpretation, or processing by human beings or computers

[SOURCE: ISO 10303-2:2024, 3.1.207]

3.6**data property**

RDF property (3.29) included in the vocabulary of an *OWL ontology* (3.20) representing a relation between *individuals* (3.12) and literal values, with semantic significance

3.7**definition**

concise statement of the meaning of an *expression* (3.11)

Note 1 to entry: For the purposes of this document definitions can be of two sorts: (1) those formulated using a natural language such as English, supplemented where necessary by technical terms or codes used in some specialist domain; (2) those formulated using a computer-interpretable language such as *OWL 2* (3.40) or common logic.

[SOURCE: ISO/IEC 21838-1:2021, 3.8]

3.8 description logic DL

family of formal knowledge representation languages that are more expressive than propositional logic but less expressive than first-order logic

[SOURCE: ISO/IEC 21972:2020(en), 3.2]

3.9 domain

class (3.4) to which the subject of an RDF statement using a given *RDF property* (3.29) belongs

Note 1 to entry: Domains and ranges of RDF properties enable inferences about the things described using those properties.

[SOURCE: Adapted from DCMI Glossary, <https://www.dublincore.org/resources/glossary/>]

3.10 entity

class (3.4) to which the subject of an RDF statement using a given *RDF property* (3.29) belong item that is perceivable or conceivable

Note 1 to entry: The term ‘entity’ is a catch-all terms analogous to ‘something’.

Note 2 to entry: Deleted second sentence in Note 1 to entry and modified term by omitting permitted term “object”.

[SOURCE: ISO/IEC 21838-1:2021, 3.1]

3.11 expression

word or group of words or corresponding symbols that can be used in making an assertion

Note 1 to entry: Expressions are divided into natural language expressions and expressions in a formal language.

[SOURCE: ISO/IEC 21838-1:2021, 3.5]

3.12 individual

particular *entity* (3.10)

Note 1 to entry: In contrast to classes or types, individuals are not exemplified or instantiated by further entities.

[SOURCE: ISO/IEC 21838-1:2019, 3.3]

3.13 information

facts, concepts, or instructions

[SOURCE: ISO 10303-2:2024, 3.1.1.55]

3.14 instance

individual (3.12) that instantiates some *class* (3.4)

EXAMPLE John, John’s laptop, the year 2012.

[SOURCE: ISO/IEC 21838-2:2021, 3.6]

3.15

internationalized resource identifier IRI

sequence of characters from the Universal Character Set (see ISO 10646) constructed according to the syntax rules provided in RFC 3987

Note 1 to entry: IRIs are generalization of URIs (ISO/IEC 19788-1:2011, 3.36). While URIs are sequence of characters chosen from a limited subset of US-ASCII, IRIs can possibly contain characters from the Universal Character Set such as Chinese, Japanese, Korean, or Cyrillic characters.

Note 2 to entry: A mapping from IRIs to URIs is defined in RFC 3987, making it possible to use IRIs instead of URIs, where appropriate, to identify resources.

[SOURCE: ISO/IEC 19788-8:2015, 3.1]

3.16

local name

reference to a local object directly accessible from a *namespace* ([3.18](#))

[SOURCE: ISO 19150-2:2015(en), 4.1.22 modified — renamed from “localName” to “local name”]

3.17

model

representation of certain *entities* ([3.10](#)) and their characteristics and relationships either (a) using a formalism, or (b) using an established or ad hoc modelling paradigm, approach, or technique

[SOURCE: ISO 15704:2019, 3.15, modified – definition: adding “and relationships”, deleted note 1 and 2 to entry]

3.18

namespace

RDF ([3.32](#)) common URI ([3.38](#)) stem used in identifiers for a set of related resources

Note 1 to entry: The RDF namespace is concatenated with the localName (4.1.22) to create the complete URI identifier for an RDF resource. Every RDF resource is identified by a URI. In contrast, an XML namespace URI scopes a local XML component name, but there is no rule for combining these into a single identifier string.

[SOURCE: ISO 19150-2:2015(en), 4.1.26, modified — deleted “prefix or”; note edited to refer to local entry]

3.19

object property

RDF *property* ([3.29](#)) included in the vocabulary of an OWL *ontology* ([3.20](#)) representing a relation between *individuals* ([3.12](#)), with semantic significance

3.20

ontology

collection of terms, relational expressions and associated natural-language definitions together with one or more formal theories designed to capture the intended interpretations of these definitions

Note 1 to entry: The following words are also found as terms in the source definition: terms, relational expressions, definitions and formal theories.

[SOURCE: ISO/IEC 21838-1:2021, 3.14]

3.21

OWL 2 ontology document ontology document

document which contains an ontology, written in one of the syntaxes of OWL 2

[SOURCE: Adapted from OWL 2 Web Ontology Language Structural Specification and Functional-Style Syntax (Second Edition), [section 3.2](#)]

3.22**OWL DL reasoning**

inference of implicit facts from knowledge expressed in an *OWL 2* (3.40) *ontology* (3.20), using the direct model-theoretic semantics of OWL 2

3.23**OWL Manchester syntax**

compact syntax for *OWL 2 ontology documents* (3.21)

Note 1 to entry: This syntax is frame-based, as opposed to the axiom-based other syntaxes for OWL 2.

[SOURCE: ...OWL 2 Manchester Syntax, <https://www.w3.org/TR/owl2-manchester-syntax/>]

3.24**pattern**

template for expressing facts using the vocabulary of an *ontology* (3.20)

3.25**prefix**

abbreviation of a *namespace* (3.18)

EXAMPLE “rdf:”, “owl:”

Note 1 to entry: RDF 1.1 Turtle specifies that “A prefixed name is a prefix label and a local part, separated by a colon ‘:’. A prefixed name is turned into an IRI by concatenating the IRI associated with the prefix and the local part”.

[SOURCE: <https://www.w3.org/TR/turtle/#prefixed-name>]

3.26**primitive**

expression (3.11) which cannot be further reduced or deconstructed

Note 1 to entry: Construct lacking necessary or sufficient conditions.

3.27**range**

class (3.4) to which the object of an RDF statement using a given *RDF property* (3.29) belongs

[SOURCE: Adapted from DCMI Glossary, <https://www.dublincore.org/resources/glossary/>]

3.28**RDF resource****resource**

IRI (3.15) or literal that denotes something in the world (the “universe of discourse”)

[SOURCE: W3C RDF 1.1, section 1.2, <https://www.w3.org/TR/2014/REC-rdf11-concepts-20140225/>]

3.29**RDF property****property**

term used to add attributes to classes and to semantically describe and derive relationships between subjects and objects

Note 1 to entry: OWL employs RDF properties such as `rdf:type` and `rdfs:subClassOf`.

Note 2 to entry: An *RDF property* (3.29) can possibly have sub-properties with restrictions on *domain* (3.9) and *range* (3.27).

Note 3 to entry: denotation of an *RDF* (3.32) predicate *IRI* (3.15).

[SOURCE: W3C RDF 1.1, section 1.2, <https://www.w3.org/TR/2014/REC-rdf11-concepts-20140225/>]

3.30

reference data library

RDL

vocabulary of *terms* (3.37) organised according to *ontological* (3.20) categories

Note 1 to entry: IDO is intended to be a suitable basic vocabulary for RDLs represented as OWL 2 ontologies.

3.31

relation

way in which *entities* (3.10) are related

Note 1 to entry: Relations can hold between individuals (this motor is part of this pump system); or between classes or types (grease is a subclass of lubricant); or between particulars and classes (this compressor is an instance of rotating equipment).

[SOURCE: ISO/IEC 21838-1:2019, 3.4, modified — in Note 1, edited text in brackets in the note to entry to make it engineering focused, replaced “particulars” with “individuals”, Note 2 to entry and Note 3 to entry removed.]

3.32

resource description framework

RDF

framework for representing *information* (3.13) in the Web as sets of triples, each consisting of a subject, a predicate and an object

[SOURCE: W3C RDF 1.1, <https://www.w3.org/TR/2014/REC-rdf11-concepts-20140225/>]

3.33

shapes constraint language

SHACL

language for validating *RDF* (3.32) graphs against a set of conditions

[SOURCE: W3C SHACL, <https://www.w3.org/TR/2017/REC-shacl-20170720/>]

3.34

tag number

unique code that identifies the equipment function and its physical location

Note 1 to entry: Normally includes the system for which it belongs.

Note 2 to entry: Also entitled “functional location” in some systems.

Note 3 to entry: The equipment is allocated to a tag for only so long as it occupies that function and location.

[SOURCE: ISO 14224:2016(en), 3.91, modified — Note 1 to entry is left out and other notes to entry renumbered accordingly. Internal cross-references deleted from Note 2 to entry and Note 4 to entry, Note 3 to entry replace “CMMIS” with “systems”]

3.35

tagged equipment

tag

equipment that is identified by a *tag number* (3.34)

3.36

taxonomy

type of hierarchy which deals with generalization or specialization relationships

[SOURCE: ISO/IEC 11179-32:2023(en), 3.20, modified — Note 1 to entry removed]

3.37**term**

expression (3.11) that refers to some *class* (3.4) or to some particular *individual* (3.12)

Note 1 to entry: An ontology will typically contain a unique ‘preferred term’ for the entities within its coverage domain. Preferred terms may then be supplemented with other terms recognised by the ontology as synonyms of the preferred terms.

[SOURCE: ISO/IEC 21838-1:2021, 3.7]

3.38**uniform resource identifier****URI**

compact sequence of characters that identifies an abstract or physical resource

Note 1 to entry: The referents of URIs in RDF data are called “resources”.

Note 2 to entry: see *IRI* (3.15) on the relationship between IRIs and URIs.

[SOURCE: IETF RFC 3986, <https://www.rfc-editor.org/info/rfc3986>]

3.39**use case**

technique for capturing potential functional requirements

Note 1 to entry: This technique employs the use of one or more scenarios to convey how the system interacts with the end user or another system to achieve one or more specific objective.

Note 2 to entry: Typically use cases treat the system as a black box, and the interactions with the system, including system responses, are as perceived from the outside of the system. Use cases are popular because they simplify the description of requirements and avoid the problem of making assumptions about how the functionality will be accomplished.

[SOURCE: IEC TS 62443-1-1:2009, 3.2.132]

3.40**web ontology language****OWL 2**

ontology language for the Semantic Web with formally defined meaning

Note 1 to entry: OWL *ontologies* (3.20) are commonly expressed in *RDF* (3.32) The structural specification of OWL is given in OWL “functional syntax”, which is “independent of the concrete exchange syntaxes for OWL 2 ontologies”.

Note 2 to entry: OWL 2 ontologies provide *classes* (3.4), properties, *individuals* (3.12), and data values and are stored as ontology documents.

[SOURCE: OWL 2 Specification, <https://www.w3.org/TR/owl2-syntax/>]

4 Abbreviated terms

BFO	basic formal ontology
BFO-ISO	ISO 21838-2 basic formal ontology
DCAT	data catalogue vocabulary
DCMI	Dublin core metadata initiative
DOLCE	descriptive ontology for linguistic and cognitive engineering
IAO	information artifact ontology

IDO	industrial data ontology
IANA	Internet Assigned Numbers Authority
IETF	internet engineering task force
IOF	industrial ontology foundry
IOF-AV	industrial ontology foundry annotation vocabulary
NPS	nominal pipe size
OBO	open biological and biomedical ontology foundry
PAV	provenance, authoring and versioning ontology
QUDT	quantities, units, dimensions, and types
RDFS	RDF schema
RDF/OWL	OWL represented in RDF
RFC	request for comments (publication by IETF)
RO	OBO relations ontology
SKOS	simple knowledge organization system
SSN	semantic sensor networks
SWRL	semantic web rule language
UCUM	unified code for units of measure in RDF
UML	unified modelling language
W3C	world wide web consortium

5 IDO entities

5.1 General

This clause documents the complete contents of the IDO ontology in OWL. A concise overview is provided for each main IDO category. Diagrams, explanatory notes and examples to illustrate how facts can be represented using the IDO vocabulary are also provided.

[Subclause 5.2](#) describes metadata for the ontology. [Subclauses 5.3](#) to [5.6](#) detail the IDO classes and relations. Each ontology resource (class or relation) is described in a subclause, with annotations (definition, cross-references, notes, etc.) and any semantic restrictions. Within these subclauses, annotation property names are abbreviated for readability; the abbreviations are listed in [Table 1](#). Details of the RDF vocabulary of annotations used in IDO is contained in [5.6](#).

Table 1 — Labels of RDF annotation properties

annotation property	entry label
iof-av:isPrimitive	primitive?
iof-av:naturalLanguageDefinition	meaning
iof-av:primitiveRationale	why primitive
iof-av:usageNote	usage note
owl:deprecated	deprecated?
rdfs:seeAlso	see also
skos:example	example
skos:scopeNote	scope note
skos:altLabel	alternative label
iof-av:explanatoryNote	explanatory note
rdfs:comment	comment
rdfs:isDefinedBy	defined by
iof-av:firstOrderLogicDefinition	first-order logic definition
iof-av:semiFormalNaturalLanguageDefinition	semi-formal definition
iof-av:semiFormalNaturalLanguageAxiom	semi-formal axiom
iof-av:adaptedFrom	adapted from
iof-av:synonym	synonym

Where appropriate, listings corresponding to an OWL ontology document are provided, using the OWL Manchester Syntax. This includes the prefix declarations [5.2.1](#), ontology declaration [5.2.2](#), and axioms used to capture semantic restrictions beyond subclass or subproperty relationships.

[Annex A](#) offers a foundational overview of modelling with IDO. [Annex B](#) describes shortcut properties, validation, and rules essential for practical applications. Additionally, [Annex C](#) outlines how the generic IDO vocabulary can be extended with specialized terminologies, illustrated by alignments with three major ontology standards.

An ontology conformant with IDO as upper ontology shall be expressed in accordance with the OWL 2 Web Ontology Language Structural Specification and Functional-Style Syntax. Additionally, it shall be consistent in accordance with OWL 2 Web Ontology Language Direct Semantics.

Diagrams are provided to aid in navigation of the class and relation taxonomies, and to illustrate modelling examples in a visual form. They follow a UML *class diagram* style and represent the RDF/OWL representation of IDO as faithfully as is practical. The diagrams are intended as an aid to the reader, not as a complete reproduction of the IDO ontology or of example models. Some contents will be left out or truncated, to avoid overly complex diagrams.

The nodes in diagrams have box shape and represent RDF resources (the *subject* and *object* of RDF triples). Arrows represent relationships (the *predicate* of RDF triples). Each box is named with a combination of *prefix*, indicating the RDF namespace of the resource, and the *local name*. Boxes are provided with circled letters to indicate OWL type of the relevant resource, as shown in [Table 2](#). Slightly extending the UML convention, arrows with open arrowheads indicate subclass or subproperty relationships (*rdfs:subClassOf* or *rdfs:subPropertyOf*, respectively); these arrows are not labelled. All other arrows are labelled with the prefixed name of the RDF resource.

Table 2 — RDF resource symbols for diagrams

symbol	resource type
(C)	OWL class
(O)	OWL object property (individual to individual relation)
(D)	OWL data property (individual to literal value relation)
(A)	OWL annotation property (non-semantic annotation)
(i)	OWL individual
(L)	RDF literal (datatype value)

Figure 2 provides the taxonomy of classes in IDO. Each node represents a class, with the prefix *lis:* indicating that the class belongs to the IDO namespace. Each arrow (with open arrowhead) represents a subclass relationship, i.e. an arrow from (sub)class *A* to (super)class *B* means every member of *A* is a member of *B*. Nodes that have no superclass assigned are implicitly subclasses of the ultimate OWL class, *owl:Thing*. See 5.3 for detailed descriptions of each class.

The IDO object properties (individual-to-individual relations) are detailed in 5.4, which contains a subclass for each subproperty of the ultimate OWL object property, *owl:topObjectProperty*. Each subclass contains a diagram that shows the property taxonomy as well as domain and range restrictions. For property nodes, an arrow with open arrowhead from (sub)property *A* to (super)property *B* means that every pair of individuals in relation *A* is also in relation *B*.

NOTE No diagrams are required for IDO data properties and annotation properties (see 5.5 and 5.6).

OWL axioms are challenging to display in diagrams that are limited to the simple node–arrow structure of RDF graphs. Axioms tend to include nested expressions, which call for anonymous RDF/OWL classes and verbose representation. For this reason, most semantic restrictions in this clause beyond simple subclass or subproperty relationships are shown only in listings.

EXAMPLE Figure 2 is an example of a diagram that does include anonymous classes (from the example accompanying 5.4.6.2). The diagram illustrates a typical IDO declaration, that the purpose of a compressor is to compress; in more formal IDO terms, that the *Function* of a *Compressor* is *realized in* only *Compression* activities. In OWL Manchester Syntax, the corresponding axiom is written as follows.

```
Class: Compressor
  SubClassOf: hasFunction some (realizedIn only Compression)
```

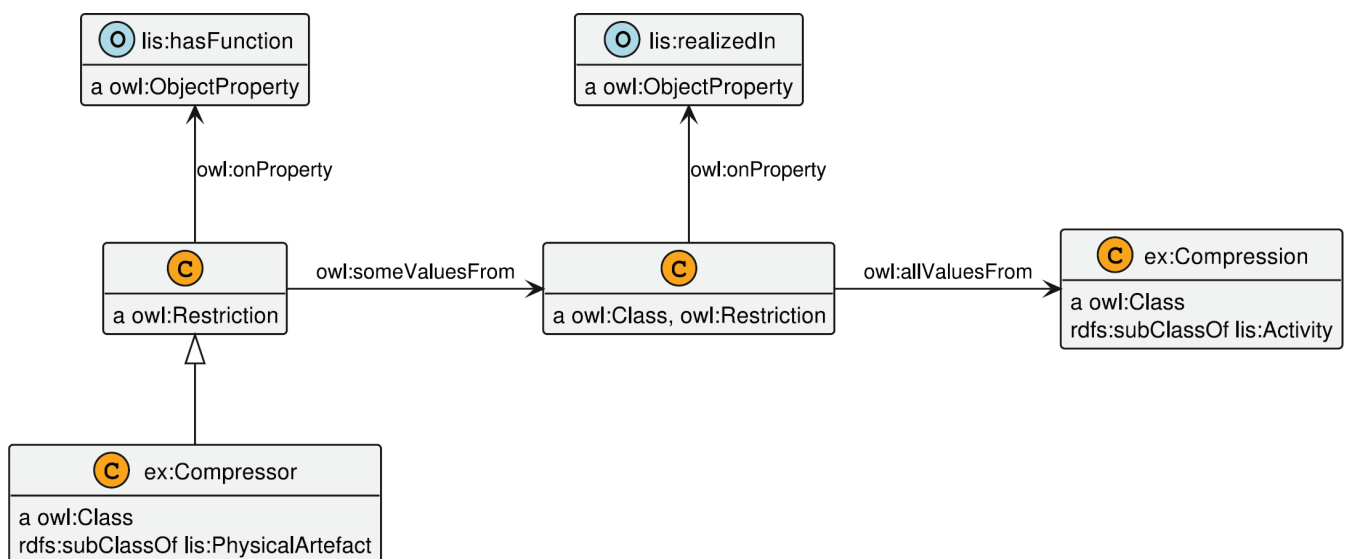


Figure 2 — IDO pattern for the function of a class of physical artefacts (here, *Compressor*)

5.2 Declarations

5.2.1 Prefixes

An IDO ontology document in OWL employs the namespace prefixes of Listing in Figure 3.

```
Prefix: lis: <https://rds.posccaesar.org/ontology/lis14/rdl/>
Prefix: owl: <http://www.w3.org/2002/07/owl#>
Prefix: rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
Prefix: xml: <https://www.w3.org/XML/1998/namespace>
Prefix: xsd: <http://www.w3.org/2001/XMLSchema#>
Prefix: rdfs: <http://www.w3.org/2000/01/rdf-schema#>
Prefix: skos: <http://www.w3.org/2004/02/skos/core#>
Prefix: pav: <https://purl.org/pav/>
Prefix: obo: <http://purl.obolibrary.org/obo/>
Prefix: dol: <http://www.loa-cnr.it/ontologies/DOLCE-Lite.owl#>
Prefix: ssn: <https://www.w3.org/ns/ssn/>
Prefix: time: <http://www.w3.org/2006/time#>
Prefix: om: <http://www.ontology-of-units-of-measure.org/resource/om-2/>
Prefix: qudt: <https://qudt.org/schema/qudt/>
Prefix: foaf: <http://xmlns.com/foaf/0.1/>
Prefix: dc: <https://purl.org/dc/elements/1.1/>
Prefix: dcterms: <https://purl.org/dc/terms/>
Prefix: dcmitype: <https://purl.org/dc/dcmitype/>
Prefix: prov: <http://www.w3.org/ns/prov#>
Prefix: iof-core: <https://spec.industrialontologies.org/ontology/core/Core/>
Prefix: iof-av: <https://spec.industrialontologies.org/ontology/core/meta/
AnnotationVocabulary/>
```

Figure 3 — Prefixes

This document contains code examples that use the prefix “ex:”. This prefix stands for “<http://example.org/>”, a domain reserved by IANA specifically for use in documentation.

5.2.2 Ontology

The URI of the IDO ontology in OWL is <https://rds.posccaesar.org/ontology/lis14/ont/core>. Figure 4 provides the ontology declaration, with metadata.

NOTE The namespace URI of the IDO ontology and contained entities is <https://rds.posccaesar.org/ontology/lis14/>, referring to an organisation external to ISO. This is in line with common practice. For example, the OWL rendering of the BFO-ISO ontology of ISO 21838-2 employs the namespace <http://purl.obolibrary.org/obo/>.

```
Ontology: <https://rds.posccaesar.org/ontology/lis14/ont/core> <https://rds.posccaesar.org/ontology/lis14/ont/core/4.1>
Annotations: rdfs:label "Industrial Data Ontology",
             dcterms:title "Industrial Data Ontology",
             owl:versionInfo "4.1 DIS December 2024",
             pav:previousVersion <https://rds.posccaesar.org/ontology/lis14/ont/core/4.0>,
             pav:hasEarlierVersion <https://rds.posccaesar.org/ontology/lis14/ont/core/3.0>,
             pav:hasEarlierVersion <https://rds.posccaesar.org/ontology/lis14/ont/core/2.0>,
             pav:hasEarlierVersion <https://rds.posccaesar.org/ontology/lis14/ont/core/1.0>,
             pav:lastUpdateOn "2024-12-10T13:38:23Z"^^xsd:dateTime ,
             dcterms:license <https://creativecommons.org/licenses/by-sa/4.0/>,
             dcterms:creator <https://orcid.org/0000-0002-7167-7321>,
```

```

dcterms:contributor <https://orcid.org/0000-0002-7167-7321>,
dcterms:contributor <https://orcid.org/0000-0002-6228-1317>,
dcterms:contributor <https://orcid.org/0000-0002-5509-8899>,
dcterms:contributor <https://orcid.org/0000-0002-9736-8316>,
dcterms:contributor <https://orcid.org/0000-0002-7336-3932>,
dcterms:contributor <https://orcid.org/0009-0005-8440-2238>,
# not issued yet dcterms:issued "2024-12-10"^^xsd:date,
dcterms:modified "2024-12-10"^^xsd:date,
dcterms:publisher <https://www.posccaesar.org>,
dc:rights "Copyright POSC Caesar Association",
rdfs:comment "The Industrial Data Ontology is an upper ontology for industrial data,
inspired by the ISO 15926-2 data model and Basic Formal Ontology.",
dcterms:description "The Industrial Data Ontology is an upper ontology for industrial
data, inspired by the ISO 15926-2 data model and Basic Formal Ontology."

```

Figure 4 — Ontology header

5.3 Classes

5.3.1 Disjointness clause

IDO serves as a foundational framework for categorizing entities within industrial domains. At its core, IDO distinguishes between three primary classes: Object, Dependent, and Temporal. Objects represent entities that exist independently and persist through time, such as process plants, personnel, or industry standards. Dependent entities, on the other hand, are inherent qualities or potentials of objects, like the colour of a surface or the storage capacity of a room. Temporal entities encompass aspects related to time, including moments, periods, activities, and events within industrial processes. This hierarchical classification system allows for a structured organization of industrial data, facilitating comprehensive modelling and analysis.

The three primary classes represent different foundational categories into which all entities are categorised. No entity may be a member of more than one of these classes. The constraint is applied by declaring the classes to be disjoint from one another.

```
DisjointClasses: lis:Temporal, lis:Object, lis:Dependent
```

5.3.2 The Object taxonomy

5.3.2.1 General

The class Object serves as one of three foundational categories within IDO, comprising a wide range of entities found within industrial environments. Objects are systematically classified into more specific subclasses, each delineating distinct types of entities encountered in industrial settings. [Figure 5](#) presents the resulting class hierarchy. The subclasses are PhysicalObject, representing tangible entities with mass and spatial occupancy, InformationObject, which captures data and information, Location, denoting spatial regions, Organisation, representing groups with common objectives, and FunctionalObject, which embodies entities with specific functions or roles. This taxonomy provides a structured framework for modelling and organizing industrial entities.

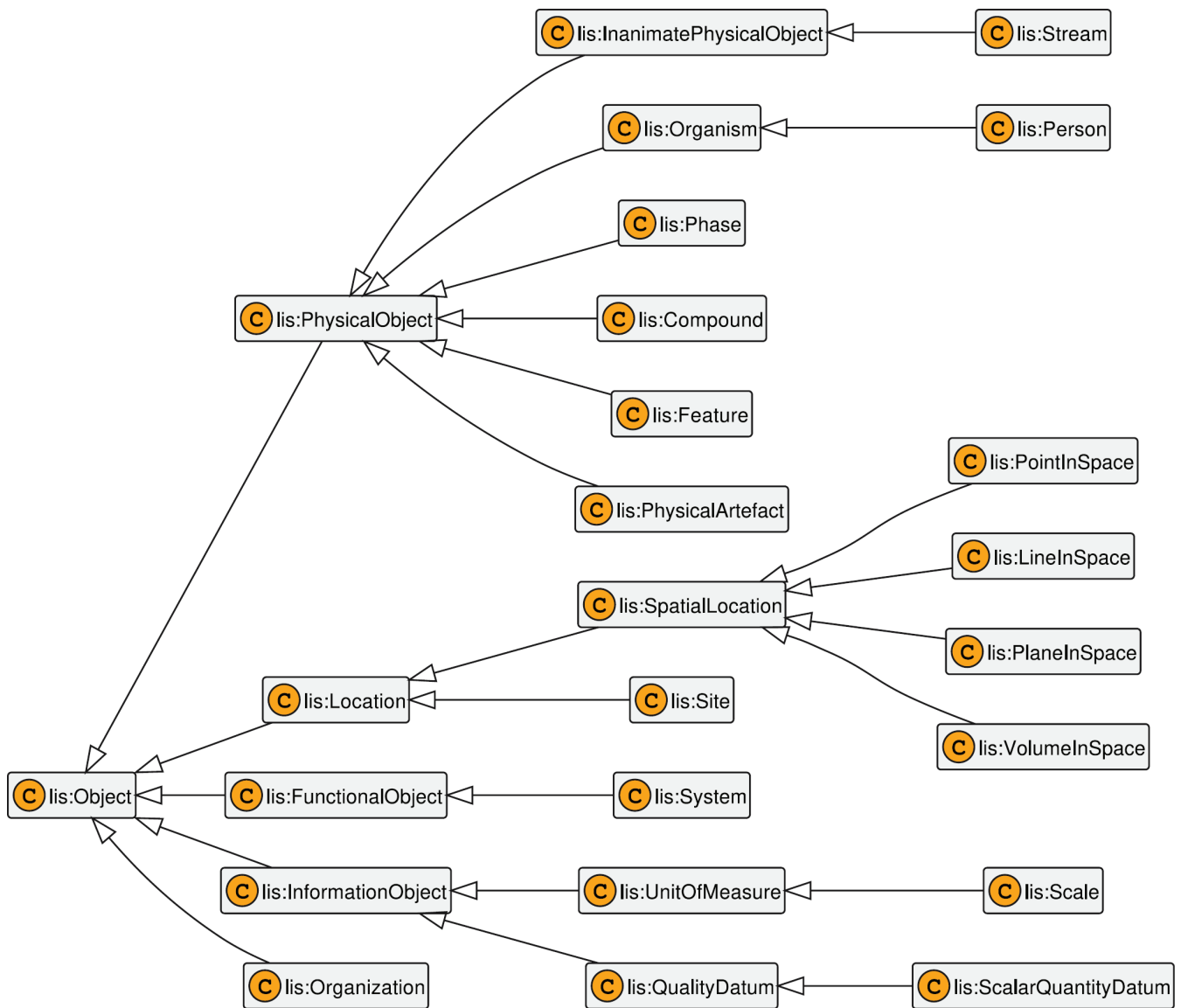


Figure 5 — IDO 'object' class hierarchy

5.3.2.2 lis:Object

- *meaning* An entity that is not dependent on other entities for existence, and which persists through time.
- *primitive?* true
- *why primitive* A precise account of what counts as an 'object' is beyond the scope of IDO.
- *example* A process plant, a compressor, a wire, a person, an operator, a database record, an industry standard, a manual document, a site.
- *see also* dol:endurant
 - *comment* From the DOLCE description of 'endurant': "The main characteristic of endurants is that all of them are independent essential wholes. ..."
- *see also* ISO 15926-2:2003 'possible_individual'
- *see also* obo:BFO_00000004
 - *comment* BFO_00000004 is 'independent continuant'.

- *scope note* In IDO, 'information object' is represented as a subclass of 'object'. According to the BFO literature, and applied in the Information Artifact Ontology and other BFO compliant ontologies, information artifacts are (multiply) dependent objects. a *sibling* class to 'independent continuant' rather than a subclass, as used in IDO.

DisjointClasses: lis:PhysicalObject, lis:InformationObject, lis:Organization, lis:Location
--

5.3.2.3 lis:FunctionalObject

Parent: lis:Object

- *meaning* An 'Object' that provides a required 'Function'.
- *primitive?* false
- *semi-formal axiom* If x is a 'functional object', then x is an 'object' that 'has function' some 'Function'.
- *example* A pump, a hammer, a password.
- *example* A password is concretized when it is entered, and it allows the computer to be unlocked. The password participates in the unlocking process through its concretization.
- *usage note* Most artefacts are both physical and functional objects as a functional object is any object that has a function.
- *usage note* Note that it is possible to model without explicitly using this class, as any object that has a function can be inferred to be a functional object.
- *usage note* This class can be used as a root class for functional classification of objects, including physical objects.
- *usage note* Artefact types tend to be described primarily in terms of 'function'; for example, the "minimal" definition of a pump would be, "a physical artefact, the function of which is to pump (to be the active participant in a pumping activity)". This kind of definition places no restriction on physical make-up and is suitable for a subclass of 'functional object'. Note that the activity type of 'pumping' is here a precursor to the definition of the physical object type 'pump'. A taxonomy of pump classes should extend from a class 'pump' defined solely in terms of function, as above.
- *usage note* A "tag", when included in an IDO asset model, should in general have a type that is subclass of both 'functional object' and 'physical artefact'.
- *see also* ISO 15926-2:2003 'functional_physical_object'
 - *comment* ISO 15926-2:2003, p. 110, states, "A functional_physical_object is a physical_object that has functional, rather than material, continuity as its basis for identity. Adjacent temporal parts of a functional_physical_object need not have common matter or energy, provided the matter or energy of each temporal part fulfils the same function."

Class: lis:FunctionalObject SubClassOf: lis:hasFunction some lis:Function
--

5.3.2.4 lis:System

Parent: lis:FunctionalObject

- *meaning* A 'functional object' with one or more functional parts whose functions contribute to its function.
- *primitive?* true
- *why primitive* A precise account of the nature of dependency between the function of a system, as a whole, and the functions of its parts is beyond the scope of IDO.

- *example* An electrical network, an HVAC system, a pumping system, a compressor.
- *usage note* Most industrial assets are both 'system' and 'inanimate physical object'. When modelling, e.g. a pump system, that system will be an 'inanimate physical object', with the pump part related to the pump system by 'has functional part'. The domain and range restrictions on 'has functional part' then imply that the pump system has type 'system', and the pump has type 'functional object'.
- *see also* lis:functionalPartOf
- *see also* ssn:System
 - *comment* The SSN definition of 'System' (2017): "System is a unit of abstraction for pieces of infrastructure that implement Procedures. A System may have components, its subsystems, which are other Systems."

```
Class: lis:System
SubClassOf: lis:hasFunctionalPart some lis:FunctionalObject
```

5.3.2.5 lis:PhysicalObject

Parent: lis:Object

- *meaning* An object that occupies space and possesses mass.
- *primitive?* true
- *why primitive* A precise account of what it means to be a physical object is beyond the scope of IDO. There is no obvious consensus on the notion of "material".
- *example* A wall, a fluid stream, a welder, a printed manual, a seafloor.
- *usage note* Installed software is considered a physical object.
- *see also* ISO 15926-2:2003 'physical_object'
 - *comment* IDO asserts that 'physical object' and 'activity' are disjunct classes. In contrast to this, and according to a 4D approach, ISO 15926-2:2003, p. 26, states that "physical_object and activity are not mutually exclusive. A possible_individual can be both".
- *see also* obo:BFO_0000040
 - *explanatory note* BFO_0000040 is 'material entity'.
- *see also* dol:physical-object

5.3.2.6 lis:InanimatePhysicalObject

Parent: lis:PhysicalObject

- *meaning* A 'physical object' that is not living.
- *primitive?* true
- *why primitive* An account of what "living" means is beyond the scope of IDO.
- *example* A tool, a building, a vehicle, an instrument, a parking lot.
- *see also* ISO 15926-2:2003 'class_of_inanimate_physical_object'
 - *comment* ISO 15926-2:2003, p. 122: "A class_of_inanimate_physical_object is a class_of_arranged_individual whose members are not living."

5.3.2.7 lis:Stream

Parent: lis:InanimatePhysicalObject

- *meaning* An inanimate physical object consisting of energy or homogeneous parts that flow through a system.
- *primitive?* true
- *why primitive* A full account of the criteria for what counts as a stream is beyond the scope of IDO.
- *example* A stream of fluid in a pipeline, electricity in an electrical wire, ore moving on a conveyor belt.
- *usage note* The class 'stream' is intended to be used for flows of material or energy, in accordance with conventional use. There is an abundance of cases where use of the term "stream" can be called into question, as in, is a stream of electricity really moving; how many cars are needed to participate before it is properly called a "flow" of cars; while a flow of sand is a stream, maybe a succession of rolling rocks would not qualify.
- *see also* ISO 15926-2:2003 'stream'
 - *comment* ISO 15926-2:2003, p. 112, states, "A stream is a physical_object that is material or energy moving along a path, where the path is the basis of identity and may be constrained. The stream consists of the temporal parts of those things that are in the stream whilst they are in it."

5.3.2.8 MateriallyClassifiedObject (lis:Compound)

Parent: lis:PhysicalObject

- *alternative label* Compound
- *meaning* A physical object with a defined material composition.
- *primitive?* true
- *why primitive* The semantics of materials, of which compounds would be a specialisation, is beyond the scope of IDO.
- *example* Types of material composition, such as 316 stainless steel, reinforced concrete, and carbon monoxide.
- *usage note* Renamed class. Checked materially in Oxford English Dictionary.
- *usage note* A typical RDL includes classes that categorise individuals according to material composition. This class is suitable as the top node of such a taxonomy.
- *scope note* Natural kinds are expressly not to be classified as 'compound'. The class 'compound' is only to be used to classify amounts of matter, with non-semantic auxiliaries used to refer to chemical composition. The semantics of natural kinds such as "helium", "steel", or "hydrocarbon" is out of scope for IDO.
- *see also* ISO 15926-2:2003 'class_of_compound'

Class: lis:Compound

SubClassOf: lis:hasQuality some lis:MaterialCompositionQuality

5.3.2.9 lis:Feature

Parent: lis:PhysicalObject

- *meaning* A physical object that is a contiguous, non-separable part of a physical object.
- *primitive?* true
- *why primitive* A precise account of what "non-separable" means is beyond the scope of IDO.

- *example* A flange face, a connection port (terminal), the inside surface of a pump, a painted area of a manifold, a bolt threading.
- *see also* ISO 15926-2:2003 'class_of_feature'
- *see also* obo:BFO_0000024
 - *explanatory note* BFO_0000024 is 'fiat object part'. This is at least very close in meaning to 'feature' as defined in in ISO 15926.
- *see also* dol:relevant-part

EXAMPLE A piping engineering project builds a catalogue of piping bulk items. The items are characterised by generic type (pipe, fitting, flange), material (alloys), and geometrical features, as specified in relevant industry standards. The designation "Weld Neck Flange NPS 2 CL150 RF WT2.77ASTM 182 Gr F55" is an example of a fully specified type of piping flange:

- "Weld Neck Flange": This term indicates that the flange is designed to be welded to a pipe or other component.
- "NPS 2": This term specifies the nominal pipe size (NPS) of the flange, the inner diameter of the pipe to which it will be welded, according to ASME B36.10
- "CL150": This term indicates the flange's pressure class rating, according to ASME B16.5
- "RF": This term stands for "raised face," which is a type of flange face that provides a seal between two flanges, according to ASME B16.5
- "WT2.77": This term indicates the wall thickness of the flange, according to ASME B36.10
- "ASTM 182 Gr F55": This term indicates the material specification for the flange, according to ASTM 182

[Figure 6](#) illustrates how a class hierarchy may be built to support the classification of an individual as belonging to this type.

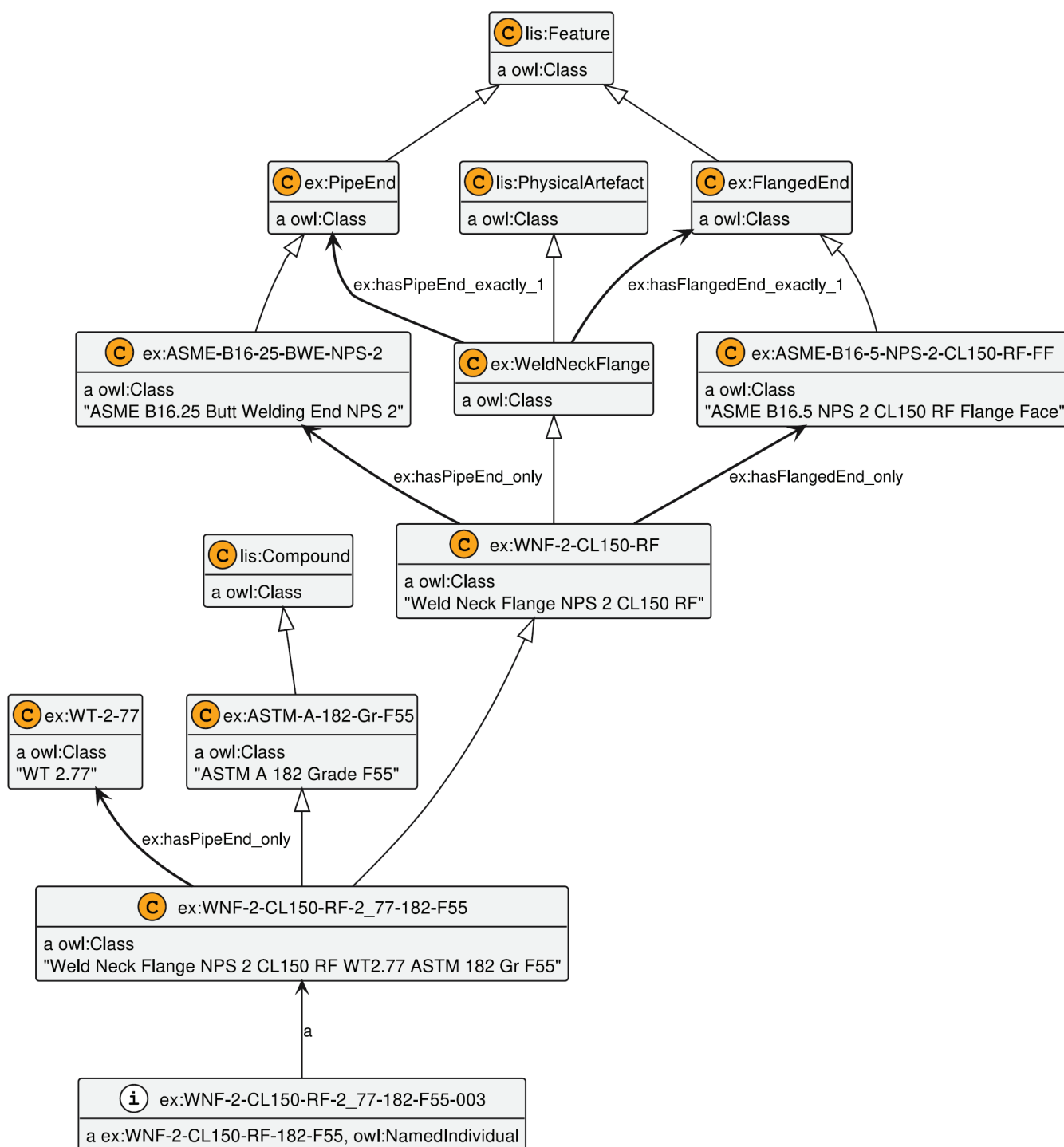


Figure 6 — 'feature' example, bulk material catalogue item

5.3.2.10 lis:Phase

Parent: `lis:PhysicalObject`

- *meaning* A physical object with a distinct and homogeneous form.
- *primitive?* true
- *why primitive* The semantics of phases is beyond the scope of IDO.
- *example* Solid object, liquid object, gaseous object.

- *usage note* Definition from Oxford English Dictionary.
- *usage note* An RDL can include classes that categorise individuals according to material phase. This class is suitable as the top node of such a taxonomy.
- *see also* ISO 15926-2:2003 'phase'

Class: `lis:Phase`
 SubClassOf: `lis:hasQuality` some `lis:PhaseQuality`

NOTE ISO 15926-2:2003, [\[2\]](#) p. 125 states that “[a] phase is a class_of_arranged_individual based on the nature of the boundary behaviour of material resulting from its atomic and molecular bonding” .

5.3.2.11 **lis:Organism**

Parent: `lis:PhysicalObject`

- *meaning* A physical object that is living.
- *primitive?* true
- *why primitive* Providing precise semantics of what “living thing” amounts to is beyond the scope of IDO.
- *example* Animal, plant, fungus, bacterium.
- *see also* ISO 15926-2:2003 'class_of_organism'

5.3.2.12 **lis:Person**

Parent: `lis:Organism`

- *meaning* An organism that is a human.
- *primitive?* true
- *why primitive* Providing precise semantics of what “human” amounts to is beyond the scope of IDO.
- *example* A human being.
- *scope note* In an RDL, subclasses of 'person' will typically require 'role' restrictions to be considered fully explicit. For example, 'maintenance engineer' will be a subclass of 'person', as every engineer is a person. To capture what a maintenance engineer does, a role restriction can say that any member is the bearer of an engineering role, and what the role amounts to will be represented by the kinds of activities that count as realizations of the role.
- *see also* ISO 15926-2:2003 'class_of_person'

5.3.2.13 **lis:PhysicalArtefact**

Parent: `lis:PhysicalObject`

- *meaning* A 'physical object' that has been created as the result of deliberate activity.
- *primitive?* true
- *why primitive* To give a precise account of “making” goes beyond the scope of IDO. An account would engage notions on which there is no clear consensus, including intentional notions of intended purpose, and of goal-directed action; of permanence and integrity; etc. There is an abundance of edge cases for “physical artefact”.
- *example* a pressure transmitter, a process plant, a wrench, a computer.

- *usage note* The superclass of 'physical artefact' is not 'inanimate physical object', in order to allow for artificially made organisms.
- *usage note* A typical RDL includes classes that represent industrial item types, such as equipment, devices, assemblies, buildings, and products, among other elements. This class is suitable as the top node in a taxonomy of such classes.

NOTE ISO 15926-4^[3] defines reference data item 25510 'artefact' as, "an inanimate physical object that is made or given shape by man."

EXAMPLE To illustrate how a 'makes' relation (as an object property) can possibly be used, consider a case in which the 'instrumentation technician' John makes a 'pressure gauge assembly'. This can be expressed in OWL with the following axioms, illustrated in [Figure 7](#).

```
Class: ex:InstrumentationTechnician
  SubClassOf: lis:Person
Class: ex:PressureGaugeAssembly
  SubClassOf: lis:PhysicalArtefact
ObjectProperty: ex:makes
Individual: ex:P-12345-PA-01
  Types: ex:PressureGaugeAssembly
Individual: ex:John
  Types: ex:InstrumentationTechnician
Facts: ex:makes ex:P-12345-PA-01
```

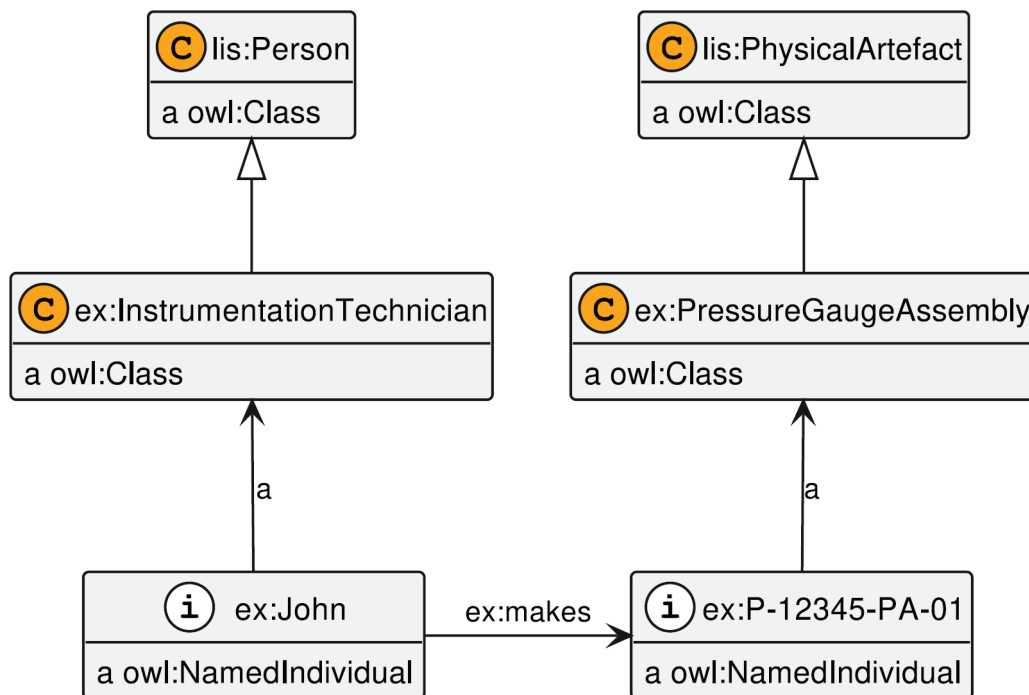


Figure 7 — 'physical artefact' example, simple pattern

For a generic 'makes' relation in an RDL, a domain of 'person' or 'organization' and a range of 'physical artefact' may be applied, with object property constraints as in the following axioms, illustrated in [Figure 8](#).

```
ObjectProperty: ex:makes
  Domain: lis:Organization or lis:Person
  Range: lis:PhysicalArtefact
```

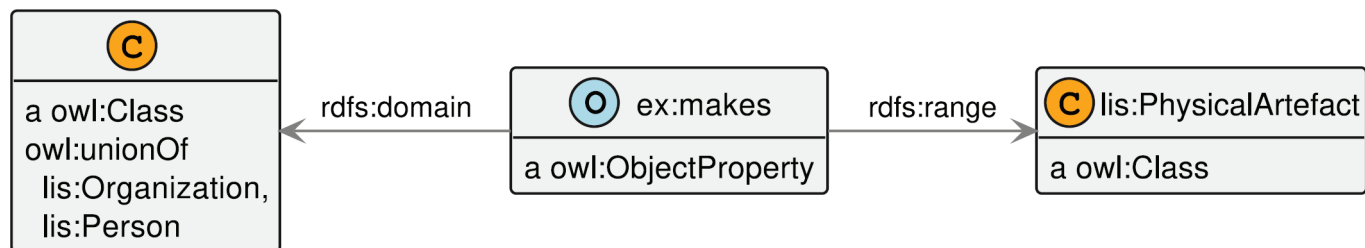


Figure 8 — 'physical artefact' example, domain and range for 'makes' example relation

A more detailed model is possible, with explicit 'role' individuals; then the 'makes' relation appears as a shortcut over the pattern of roles and created output of activities, as in the following axioms, illustrated in [Figure 9](#). See the example accompanying [5.3.3.7](#) for details on 'maker role'.

```

Class: ex:Assembling
Class: ex:InstrumentMakerRole
Individual: ex:MakingP12345-PA-02
  Types: ex:Assembling
  Facts: lis:creates ex:P-12345-PA-01
Individual: ex:JohnsInstrumentMakerRole
  Types: ex:InstrumentMakerRole
  Facts: lis:realizedIn ex:MakingP12345-PA-02
Individual: ex:John
  Facts: ex:makes ex:P-12345-PA-01, ex:hasMakerRole ex:JohnsInstrumentMakerRole

```

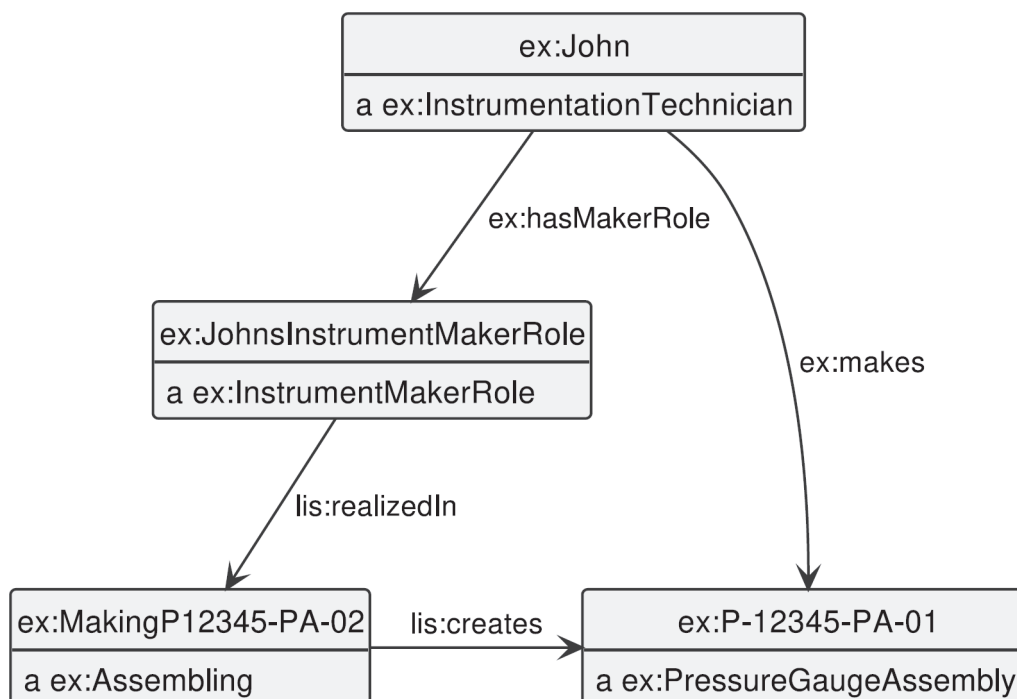


Figure 9 — 'physical artefact' example, pattern with role

The detailed model motivates adding a property chain axiom to ensure 'makes' relationships are inferred from the detailed pattern, expressed in the following axioms and illustrated in [Figure 10](#) (extending on [Figure 8](#)).

```
ObjectProperty: ex:hasMakerRole
ObjectProperty: ex:makes
SubPropertyChain: ex:hasMakerRole o lis:realizedIn o lis:creates
```

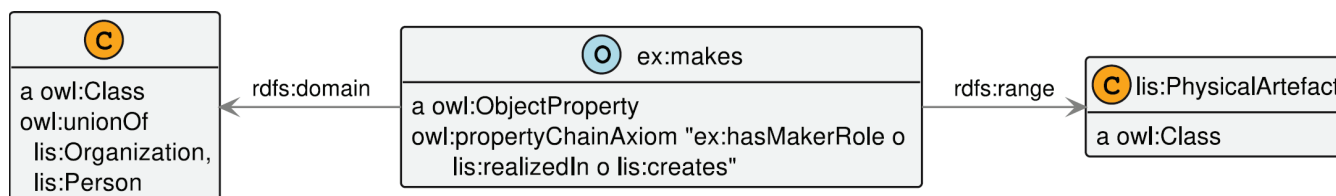


Figure 10 — 'physical artefact' example, property path for 'makes' example relation

5.3.2.14 lis:InformationObject

Parent: lis:Object

- *meaning* An object that captures information about something.
- *primitive?* true
- *why primitive* The notion of aboutness is not defined.
- *example* A report, a drawing, a P&ID diagram, a purchase order, a certificate, a database, a database record, a measurement datum, a unit of measure, an identifier, an algorithm, a programming language.
- *usage note* A common application of 'information object' is in representing facts about the vocabulary of the ontology itself. The OWL facility of "punning" treats OWL classes and properties as OWL individuals, enabling OWL object properties to be applied to them. In use cases that involve punning, the SKOS vocabulary is recommended to be used, with the SKOS classes 'concept', 'concept scheme', and 'collection' as subclasses of 'information object'.
- *scope note* The BFO class 'generically dependent continuant' represents the notion of a pattern that exists in at least one copy, e.g. "the sequence of this protein molecule". This is more general than 'information object', and not at present included in IDO. 'information object' is a subclass, with the added condition that the pattern 'is about' something.
- *see also* obo:IAO_0000030
 - *comment* IAO_0000030 is 'information content entity', a subclass of BFO_0000031 'generically dependent continuant' defined in the Information Artifact Ontology.
- *see also* obo:BFO_0000031
 - *comment* BFO_0000031 is 'generically dependent continuant'.
- *see also* ISO 15926-2:2003 'class_of_information_object'

NOTE On OWL "punning", see Krötzsch et al., [\[4\]](#) 8.3 "Entity Declarations".

5.3.2.15 lis:QualityDatum

Parent: lis:InformationObject

- *meaning* An information object that quantifies a quality.

- *primitive?* true
- *why primitive* A precise account of the representation of magnitudes is beyond the scope of IDO.
- *example* 10 volts, high (of pressure), low (of temperature), category 3 (as used to rate the strength 'quality' of a storm), 3 MB (a file size), qualifications, grades.
- *usage note* Quantification may be on nominal, ordinal, and other scales.
- *usage note* This class may be used for data that don't fit the restriction to "single unit, single number" of 'scalar quantity datum', including data employing vectors instead of numbers, and conventional categories like "high" and "medium".
- *scope note* Introduce subclasses of 'quality datum' in RDL to distinguish categories of data, such as nominal versus measured values.
- *comment* This class is inspired by the class "measurement datum" of the Information Artifact Ontology (IAO). The change of wording from "measurement" to "quality" is intended to support cases where measurement is not involved, such as with nominal values.
- *see also* obo:IAO_0000109
 - *comment* IAO_0000109 is 'measurement datum', a class in the Information Artifact Ontology.

Class: `lis:QualityDatum`
 SubClassOf: `lis:quantifiesQuality` some `lis:Quality`

5.3.2.16 `lis:ScalarQuantityDatum`

Parent: `lis:QualityDatum`

- *meaning* A quality datum that quantifies a physical quantity with a numeric value and a scale.
- *primitive?* false
- *semi-formal axiom* If x is a 'scalar quantity datum', then x is a 'quality datum' which 'quantifies quality' some 'physical quantity', and has a 'datum uom' relationship to a 'unit of measure' and a 'datum value' relationship to a literal value.
- *example* 10 kilograms, 220 volt.
- *usage note* A scalar quantity datum has a unique unit of measure and a unique numeric value. This class is inspired by the class "scalar measurement datum" of the Information Artifact Ontology.
- *see also* obo:IAO_0000032
 - *comment* IAO_0000032 is 'scalar measurement datum', a class in the Information Artifact Ontology.
- *see also* om:Measure
 - *comment* OM 2 describes 'measure' as follows. "A measure combines a number to a unit of measure. For example, '3 m' is a measure."
- *see also* qudt:QuantityValue
 - *comment* QUDT describes 'quantity value' as follows. "A Quantity Value expresses the magnitude and kind of a quantity and is given by the product of a numerical value n and a unit of measure U. The number multiplying the unit is referred to as the numerical value of the quantity expressed in that unit."

```

Class: lis:ScalarQuantityDatum
  SubClassOf: lis:datumUOM some lis:UnitOfMeasure,
    lis:datumValue some rdfs:Literal,
    lis:quantifiesQuality some lis:PhysicalQuantity

```

5.3.2.17 lis:UnitOfMeasure

Parent: lis:InformationObject

- *meaning* An information object that defines the magnitude of a 'quality', adopted by convention or by law.
- *primitive?* true
- *why primitive* Precisely determining what qualifies as a unit of measure is beyond the scope of IDO.
- *example* carton, cup, temperature degree, instances of scales
- *usage note* 'unit of measure' may be used when there is ambiguity about the scale of the measure.
- *see also* om:Unit
 - *comment* 'unit' is a class in the Ontology of units of Measure (OM 2).
- *see also* obo:UO_0000000
 - *comment* UO_0000000 is 'unit', a class in the OBO Units of measure ontology.

5.3.2.18 lis:Scale

Parent: lis:UnitOfMeasure

- *meaning* A unit of measure that, when paired with a number, unambiguously describes the magnitude of a physical quantity.
- *primitive?* true
- *why primitive* A comprehensive definition of the notion of scale is beyond the scope of IDO.
- *example* centimetre, kilogram per minute, degree Celsius, lunar distance, kilo-lines of code
- *scope note* Arguably, IDO conflates the notions of "unit of measure" and "scale" in an unhealthy way. While a unit of measure is a definite magnitude, a scale arises from fixing a number range to express multiples of the unit of measure. One may argue that a unit of measure has an implicit scale, obtained by multiplying the unit by any number; this will then allow for undefined or meaningless expressions like "minus 10 Kelvin".
- *usage note* 'scale' corresponds to the common sense notion of unit of measure which unambiguously defines the magnitude of the measure.
- *usage note* OWL data ranges can capture the acceptable numerical range for a scale. For instance, the 'datum value' of any 'quality datum' that has Celsius as 'datum uom' can be limited to the range from -273.15 upwards to infinity.
- *see also* qudt:Scale
 - *comment* QUDT provides the following subclasses of 'scale': 'Enumeration scale', 'Interval scale', 'Nominal scale', 'Ordinal scale', 'Ratio scale'. Note that the notion of nominal scales diverges in being a coding scheme, with numbers not implying relative magnitude.
- *see also* om:Scale
- *see also* ISO 15926-2:2003 'scale'

5.3.2.19 lis:Organization

Parent: lis:Object

- *meaning* An object comprising one or more persons who identify by some name and pursue a common set of plans and objectives.
- *primitive?* true
- *why primitive* A comprehensive definition of the notion of organization is beyond the scope of IDO (it would determine which kinds of aggregations of people, and which kinds of roles, are appropriate for organizations).
- *example* A standards body, a service team, a workgroup, a company
- *adapted from* iof-core:Organization
 - *comment* Organization is a class in the IOF Core ontology.
- *see also* ISO 15926-2:2003 'class_of_organization'

5.3.2.20 lis:Location

Parent: lis:Object

- *meaning* An object that is a region in space.
- *primitive?* true
- *why primitive* A comprehensive definition of the notion of location is beyond the scope of IDO.
- *example* An outdoor plant area, an opening in a wall, the space inside an enclosure
- *usage note* IDO is an anthropocentric ontology in the sense that it disregards complexities of the deep theories of the physical sciences that have only negligible impact at the standard industrial scale. For example, this means ignoring cosmological effects on the average industry and adopting a geocentric model of the universe where there exists a commonly agreed upon stationary earthly "spatial reference system".
- *see also* ISO 15926-2:2003 'spatial_location'
- *see also* dol:space-region

DisjointClasses: lis:Site, lis:SpatialLocation

5.3.2.21 lis:Site

Parent: lis:Location

- *meaning* A location determined in relation to one or more physical objects.
- *primitive?* true
- *why primitive* A comprehensive definition of the notion of demarcation is beyond the scope of IDO.
- *example* A flight deck, the inside of a manifold, the hold of a ship, a centre of gravity, an outside area
- *scope note* The position of a 'site' follows the position of the 'physical object' by which it is demarcated.
- *scope note* Locations such as countries and cities are also considered sites.
- *usage note* The utility of a 'site' may be represented by assigning one or more 'role's, such as 'front desk role', 'storage space role', or 'entryway role'.

- *comment* This class is inspired by the BFO class 'site'.
- *see also* obo:BFO_0000029
 - *comment* BFO_0000029 is 'site'
- *see also* ISO 15926-2:2003 'spatial_location', 'region_in_space'

5.3.2.22 **lis:SpatialLocation**

Parent: lis:Location

- *meaning* A location defined relative to a spatial reference system.
- *primitive?* true
- *why primitive* To provide an account of space, or spatial reference system, is beyond the scope of IDO.
- *example* The cube-shaped region extending 1 metre in each direction from origo, the line extending 10 metres from origo along the z axis, two points with coordinates on a two-dimensional plane; the location of a container at 10:00 AM.
- *usage note* As with 'spatiotemporal region', a spatial location need not be connected; for example, a location may consist in several cube-shaped regions.
- *usage note* The subclasses 'point in space', 'line in space', 'plane in space', and 'volume in space' are all defined as singular: for example, a 'point in space' individual may not have multiple points as parts. This, in contrast to the analogous BFO classes 'one-dimensional spatial region', etc., which are all defined as plural, allowing for disconnected parts.
- *scope note* The position of a 'spatial location' is fixed (relative to the frame of reference); it is not dependent on changing positions of 'physical object's.
- *see also* obo:BFO_0000006
 - *comment* BFO_0000006 is 'spatial region'.
- *see also* ISO 15926-2:2003 'spatial_location', 'region_in_space'

5.3.2.23 **lis:PointInSpace**

Parent: lis:SpatialLocation

- *meaning* A spatial location that has zero dimension with no proper parts.
- *primitive?* true
- *why primitive* A comprehensive account of geometries is beyond the scope of IDO.
- *example* The location of a probe at a given point in time

5.3.2.24 **lis:LineInSpace**

Parent: lis:SpatialLocation

- *meaning* A spatial location consisting of one or more lines.
- *primitive?* true
- *why primitive* A comprehensive account of geometries is beyond the scope of IDO.
- *example* A line connecting two points with coordinates relative to the frame of reference, the path of a flexible pipe at 12:00 AM.

5.3.2.25 lis:PlaneInSpace

Parent: lis:SpatialLocation

- *meaning* A spatial location that is a two-dimensional plane.
- *primitive?* true
- *why primitive* A comprehensive account of geometries is beyond the scope of IDO.
- *example* The midway horizontal plane of a volume given by coordinates, the plane of a moving work platform at 12:00

5.3.2.26 lis:VolumeInSpace

Parent: lis:SpatialLocation

- *meaning* A spatial location that is a three-dimensional volume.
- *primitive?* true
- *why primitive* A comprehensive account of geometries is beyond the scope of IDO.
- *example* A volume designated for 3D engineering at a point in time, a box-shaped volume determined by coordinates in a 3D frame of reference; the extent of a gas leak at 12:00

NOTE This class has been renamed from “RegionInSpace” to better align with the use of “region” in the ‘temporal’ taxonomy.

5.3.3 The Dependent taxonomy**5.3.3.1 General**

The class Dependent serves as one of three foundational categories within IDO, alongside Object and Temporal. Dependent entities are those that rely on other entities for their existence or properties. This class is further divided into two main subclasses: Potential and Quality. Potential entities represent attributes or potentials that can be realized through participation in activities, such as the sealing function of a gasket or the disposition of a material to corrode. On the other hand, Quality entities denote inherent attributes present whenever their bearer exists, encompassing properties like temperature, strength, or colour. Further subclasses are introduced resulting in the class hierarchy presented in [Figure 11](#).

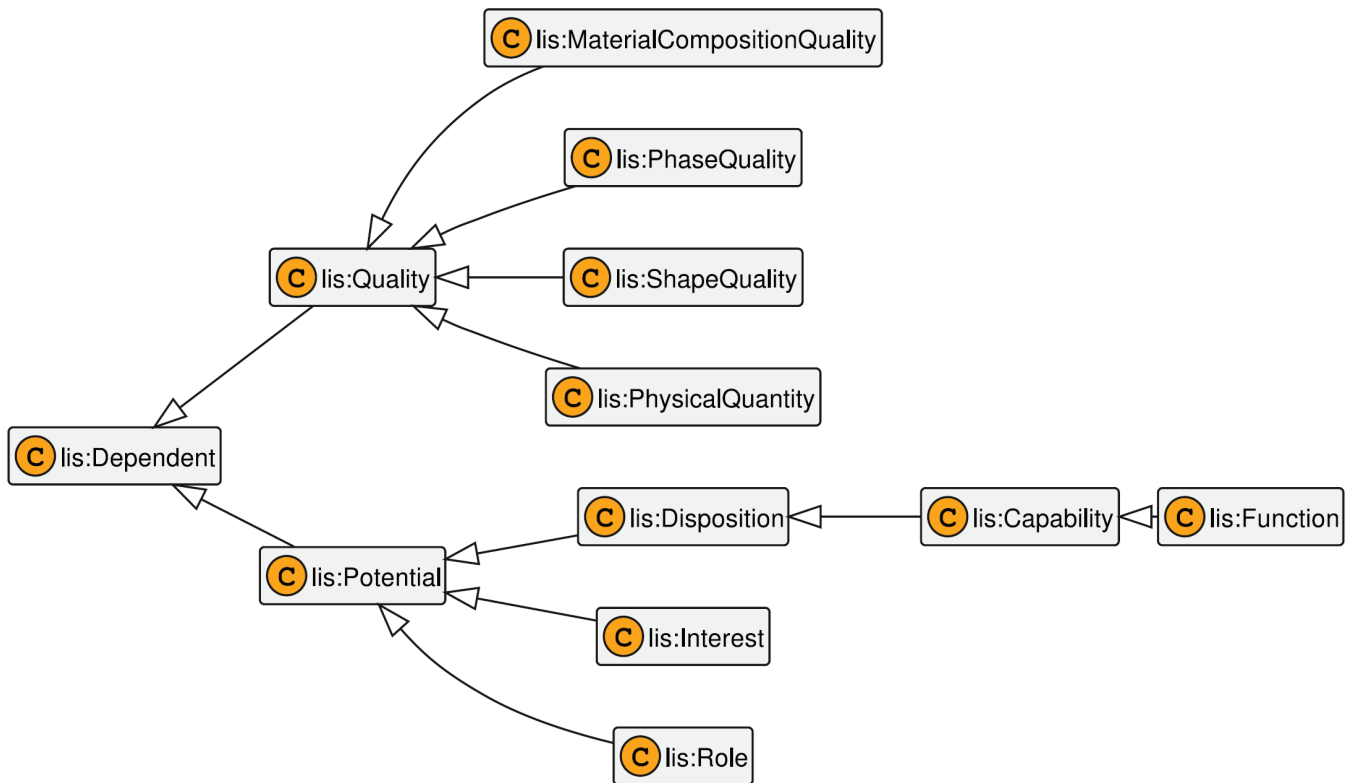


Figure 11 — IDO 'dependent' class hierarchy

5.3.3.2 lis:Dependent

- *alternative label* Dependent Entity
- *meaning* An entity that is an inherent quality or realizable potential of an object.
- *primitive?* true
- *why primitive* Providing necessary and sufficient conditions for dependence is beyond the scope of IDO. Note that the BFO analogue, 'specifically dependent continuant', is defined in terms of the primitive relation 'specifically depends on' (s-depends on), which is not included in IDO.
- *example* The colour of a surface, the temperature of a medium, a technical contact role, the storage-space role of a room.
- *scope note* Any 'object' will be the bearer of various 'dependent' entities, but any 'temporal' entity has only an indirect relationship to 'dependent's. This is in line with BFO.
- *usage note* A common need is to relate a measure based on a 'physical quantity' to an 'activity', like average speed to a transport activity. This may not be implemented with an 'average speed' entity that is dependent on the transport. For a suitable representation pattern, see the class 'activity profile' and related resources. A "shortcut" approach may use the relation 'has participant quality quantified as'.
- *see also* lis:ActivityProfile
- *see also* lis:hasParticipantQualityQuantifiedAs
- *comment* The class Dependent is closely related to "specifically dependent continuant" in BFO (BFO_0000020).

- *adapted from* obo:BFO_0000020
 - *comment* BFO_0000020 is 'specifically dependent continuant'.
- *see also* ISO 15926-2:2003 'property', 'functional_physical_object'

DisjointClasses: lis:Potential, lis:Quality

5.3.3.3 lis:Potential

Parent: lis:Dependent

- *alternative label* Realizable Entity
- *meaning* A dependent entity that can potentially be realised through participation in an activity.
- *primitive?* true
- *why primitive* The necessary condition for being a 'potential' is that activity type and participation type are provided. It is however not clear what sufficient conditions would consist in, as these would at least involve a notion of successful execution for the relevant activities, which goes beyond the scope of IDO.
- *example* The sealing function of a gasket, the disposition of a steel bar to corrode.
- *usage note* Use OWL property chains to enforce constraints on how the bearer needs to participate in order for an activity to count as a realization.
- *adapted from* obo:BFO_0000017
 - *comment* BFO_0000017 is 'realizable entity'.

NOTE The IDO vocabulary of potential (realizable) entities closely follows the account of Arp and Smith^[5].

5.3.3.4 lis:Disposition

Parent: lis:Potential

- *meaning* A potential whose realization is dependent on an object's structure.
- *primitive?* true
- *why primitive* Making the notion of "dependence on an object's structure" fully precise is beyond the scope of IDO.
- *example* The disposition of a steel bar to corrode, the disposition of a gasket to leak. The disposition to overflow a memory buffer.
- *scope note* An object may not lose a disposition without undergoing physical change.
- *adapted from* obo:BFO_0000016
 - *comment* BFO_0000016 is 'disposition'.
- *see also* lis:hasDisposition

5.3.3.5 lis:Capability

Parent: lis:Disposition

- *meaning* A disposition which when realized serves a positive or desired outcome.
- *primitive?* true
- *why primitive* Making the notion of "positive interest" fully precise is beyond the scope of IDO.

- *example* The capability of a motor to generate power, of a pump to generate head and flow. The capability to use a hammer as a paperweight, which is not its primary capability (function).
- *see also* lis:hasCapability
- *see also* obo:SIO_000112
 - *comment* SIO_000112 is 'capability'.
- *see also* obo:MF_0000043
 - *comment* MF_0000043 is 'capability'.
- *see also* obo:NCIT_C74979
 - *comment* NCIT_C74979 is 'capability'.

NOTE The definition of this class is intended to follow Merrell et al.,^[6] p. 13, where the term “capability” is defined in terms of “interest in”, specifically as a “disposition in whose realization someone (or some organism, or some group of organisms) has or had an interest”. The term ‘capability’ appears as a class in 6 different OBO Foundry ontologies, with identifiers obo:SIO_000112, obo:MF_0000043, and obo:NCIT_C74979.

5.3.3.6 lis:Function

Parent: lis:Capability

- *meaning* A capability that meets a design requirement.
- *primitive?* true
- *why primitive* Making the notion of meeting a requirement fully precise is beyond the scope of IDO.
- *example* The function of a motor to apply a specified torque to a shaft.
- *usage note* The typical ‘function’ of a hammer is realized when the hammer participates as the tool in a successful nail-driving activity. If the hammer has multiple functions, such as extracting nails using a claw or shaping objects with a ball, each function should be represented with a distinct individual.
- *see also* lis:hasFunction
- *adapted from* obo:BFO_0000034
 - *comment* BFO_0000034 is ‘function’.
- *see also* ISO 15926-2:2003 ‘functional_physical_object’

5.3.3.7 lis:Role

Parent: lis:Potential

- *meaning* A potential that is assigned to an object.
- *primitive?* true
- *why primitive* Making the notion of assignment (to a task) fully precise is beyond the scope of IDO. With complex intentions notions of obligation, which are often required in role assignments, this is particularly clear.
- *example* The role an individual person takes in a work situation, such as a manager. A student role.
- *example* The role of an interior space (a ‘site’) as a meeting room is realized when it serves as the space for conducting meetings. The site may be assigned a storage room role at a different time.

- *usage note* As with any 'potential', the existence of a role is dependent on its bearer. A particular meeting room's role may therefore not be transferred to a different room; a different role individual with similar classification would be needed.
- *scope note* This class is intended to serve the same purpose as the ISO 15926-2:2003 'role' entity type, as well as the same-named BFO class (BFO_0000023).
- *see also* lis:hasRole
- *adapted from* obo:BFO_0000023
 - *comment* BFO_0000023 is 'role'.
- *see also* ISO 15926-2:2003 'role', 'role_and_domain'

EXAMPLE To represent that a manufacturer creates a line of products, or a craftsperson fabricates one-off items, a 'maker role' can be introduced. We stipulate that a 'maker role' is realized only in a 'making' activity which 'has active participant' the role-bearer. The 'activity' class 'making' is introduced as a prerequisite. This may be expressed with the following axioms, illustrated in [Figure 14](#).

```
Class: ex:Making
  SubClassOf: lis:Activity, lis:creates some lis:PhysicalArtefact

Class: ex:MakerRole
  SubClassOf: lis:Role, lis:realizedIn only ex:Making
```

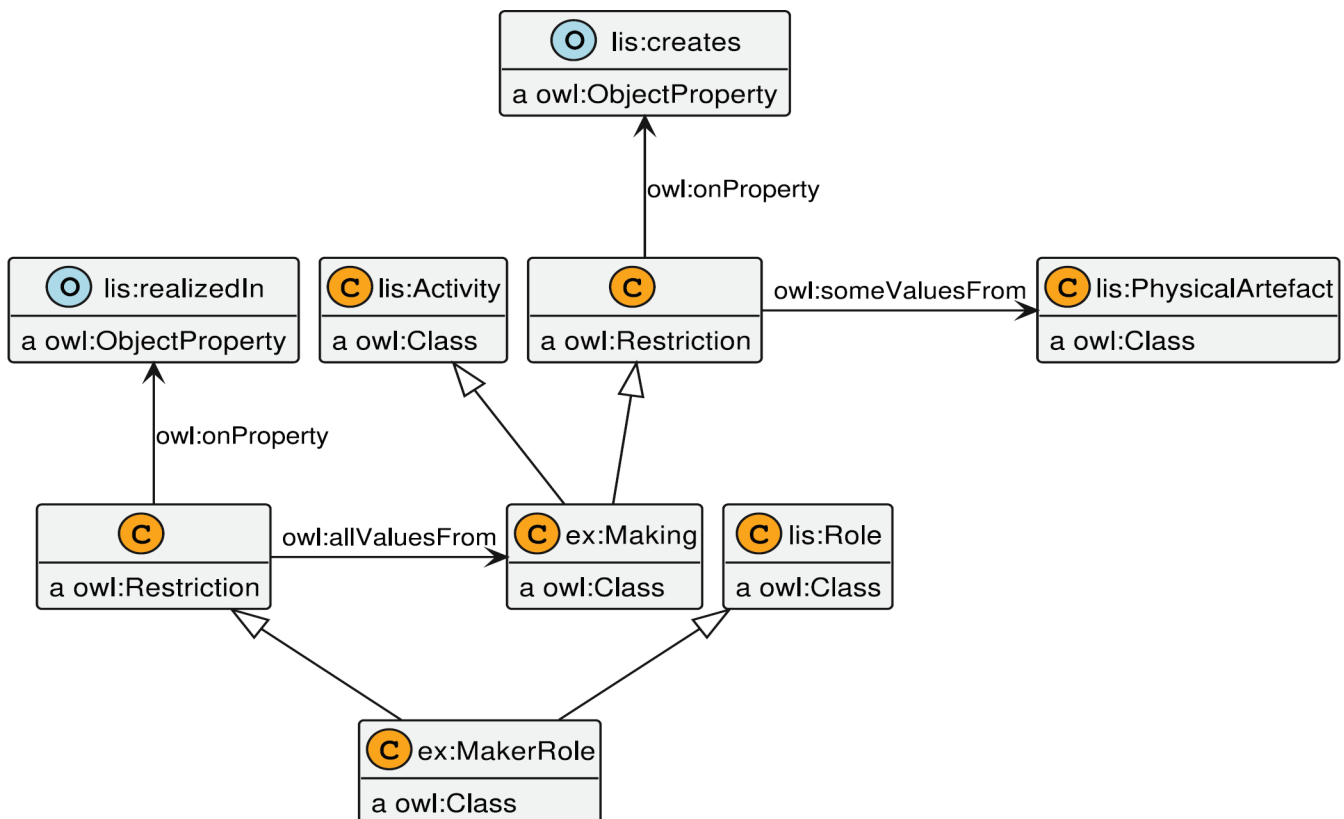


Figure 12 — 'role' example, 'maker role'

5.3.3.8 **lis:Interest**

Parent: **lis:Potential**

- *meaning* A potential, the realization of which is of value for the bearer.
- *primitive?* true
- *why primitive* Accounting for (mental) attitudes of value is beyond the scope of IDO.
- *example* A plan to act, an approval of a survey.

5.3.3.9 **lis:Quality**

Parent: **lis:Dependent**

- *meaning* A dependent entity that is present any time its bearer exists.
- *primitive?* true
- *why primitive* Qualities cover a broad range of what is commonly referred to as “properties”. It is beyond the scope of IDO to precisely determine what should be included.
- *example* The temperature of a fluid stream, the tensile strength of a bolt, the length of a pipeline, the colour of a surface coating, the familiarity of a specialist with a modification procedure, the quality of a window of being open or closed.
- *usage note* Qualities should only be used for root properties such as voltage, pressure, etc. Where there are variants such as nominal, rated, actual voltage, modellers should use **lis:Quality** for voltage and use a subclass of **lis:QualityDatum** for the moderator (“nominal”, etc.)
- *usage note* Qualities are in general subject to change over the lifetime of the bearer. The main IDO device for representing quality changes is through ‘datum’ individuals: see **lis:qualityQuantifiedAs**.
- *see also* ISO 15926-2:2003 ‘property’
 - *explanatory note* In ISO 15926-2:2003, the term “quality or characteristic” is used to explain what the entity type ‘property’ represents (clause 5.2.26.5, notes 1 and 2). Note 2 reads, “The types of characteristic or quality, such as temperature or density, are instances of class_of_property.”
- *comment* The class **Quality** and its subclass **PhysicalQuantity** are inspired by corresponding classes included in the DOLCE and BFO upper ontologies.
- *see also* obo:BFO_0000019
 - *comment* BFO_0000019 is ‘quality’.
- *see also* obo:PATO_0000001
 - *comment* PATO_0000001 is ‘quality’.
- *see also* obo:PATO_0001241
 - *comment* PATO_0001241 is ‘physical object quality’, defined in the Information Artifact Ontology as “A quality which inheres in a continuant”.
- *see also* dol:quality
 - *comment* DOLCE describes qualities as follows: “Qualities can be seen as the basic entities we can perceive or measure: shapes, colours, sizes, sounds, smells, as well as weights, lengths, electrical charges... ‘Quality’ is often used as a synonymous of ‘property’, but this is not the case in this upper ontology: qualities are particulars, properties are universals. Qualities inhere to entities: every

entity (including qualities themselves) comes with certain qualities, which exist as long as the entity exists.”

- *see also* `ssn:Property`
- *comment* SSN defines ‘Property’ as follows (2017): “A quality of an entity. An aspect of an entity that is intrinsic to and cannot exist without the entity.”
- *see also* `ssn:ActuatableProperty`
- *comment* SSN offers the following as an example for `ActuatableProperty`, a subclass of `Property`: “A window actuator acts by changing the state between a frame and a window. The ability of the window to be opened and closed is its `ActuatableProperty`.”

NOTE On relational qualities, like for “delta” pressure, see Arp et al.,^[7] p. 97 f.

5.3.3.10 `lis:PhysicalQuantity`

Parent: `lis:Quality`

- *meaning* A quality that can be quantified by measurement.
- *primitive?* true
- *why primitive* It is beyond the scope of IDO to give an account of measurement.
- *example* The mass of a bar of steel, the temperature of a medium, the inner diameter of a pipe, the lift height of a crane, the head (lift height) of a pump.
- *see also* ISO 15926-2:2003 ‘property’
- *see also* `obo:PATO_0001018`
 - *comment* `PATO_0001018` is ‘physical quality’, defined in the Information Artifact Ontology as “A quality of a physical entity that exists through action of continuants at the physical level of organisation in relation to other entities.” Note that this is distinct from ‘physical object quality’; see annotation to `lis:Quality`.
- *see also* `dol:physical-quality`
 - *comment* DOLCE defines ‘physical-quality’ as, “A quality inherent in a physical endurant.”
- *see also* `ssn:ObservableProperty`
 - *comment* SSN offers the following examples for `ObservableProperty`, a subclass of `Property`: “The height of a tree, the depth of a water body, or the temperature of a surface are examples of observable properties, while the value of a classic car is not (directly) observable but asserted.”
- *see also* `qudt:QuantityKind`
 - *comment* QUDT explains ‘quantity kind’ as, “any observable property that can be measured and quantified numerically. Familiar examples include physical properties such as length, mass, time, force, energy, power, electric charge, etc. Less familiar examples include currency, interest rate, price to earning ratio, and information capacity.”
- *see also* `om:Quantity`
 - *comment* OM2 explains ‘quantity’ as, “A quantity is a representation of a quantifiable (standardised) aspect (such as length, mass, and time) of a phenomenon (e.g. a star, a molecule, or a food product). Quantities are classified according to similarity in their (implicit) metrological aspect, e.g. the length of my table and the length of my chair are both classified as length.”