GUIDE 77-2

Guide for specification of product properties and classes

Part 2:
Technical principles and guidance

First edition 2008

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Contents

Page

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.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

Draft Guides adopted by the responsible Committee or Group are circulated to the member bodies for voting. Publication as a Guide requires approval by at least 75 % of the member bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights.

ISO/IEC Guide 77-2 was prepared by the Joint Technical Advisory Group of the ISO Technical Management Board and the IEC Standardization Management Board on product properties and families.

ISO/IEC Guide 77 consists of the following parts, under the general title *Guide for specification of product properties and classes*:

— *Part 1: Fundamental benefits*

— *Part 2: Technical principles and guidance*

— *Part 3: Experience gained*

Introduction

The capability to characterize products in an abstract way, independently of any particular manufacturer, is a fundamental aspect of engineering knowledge. Such a characterization is done by the name of a category of products that fulfils the same function, e.g. ball bearing, screw, capacitor. Such a category is called a characterization class. This first level of characterization is further detailed by means of some property-value pairs, which describe more precisely the target product within its characterization class. Examples of such properties are inner diameter, threaded length and capacitance.

Depending upon the context, products need to be characterized at various levels of accuracy. For example, in a preliminary engineering design phase, the need for a bearing with a particular inner diameter could be defined, with other properties of the bearing left unspecified. At a latter stage, it could be decided to use a ball bearing with a particular outside diameter and ball diameter; finally, a thrust angular contact ball bearing could be selected with contact angle equal to 80.

Consequently, characterization classes need to be defined at various levels of generality. A specialized class makes it possible to refine a requirement described in terms of a more general class. This generalization/specialization relationship between characterization classes builds up a hierarchy of characterization classes where each general class is refined into more specialized classes. The most specialized classes, called the leaf classes, are at the bottom of the hierarchy. In such a hierarchy, each class is associated with properties that can be used to characterize all the products belonging to this class. Note that a ball bearing being a bearing, all the properties applying to a bearing also apply to a ball bearing. Properties defined for more general classes can therefore also be used for all their specialized classes. Applicability of properties to classes is said to be inherited over the hierarchy.

Neither product characterization classes nor product properties are single words. They are concepts that belong to the engineering knowledge of specific engineering domains. These concepts exist independently of any particular language, and they can be referenced in various languages and technical jargons using various words. Usual word dictionaries are therefore inappropriate for describing such multilingual knowledge and for making it computer interpretable.

The goal of ISO/IEC Guide 77 is to describe how this knowledge can be modelled in a language-independent and computer interpretable way.

For each technical domain, the target result essentially consists of:

— a hierarchy (or a set of hierarchies) of characterization classes, each associated with a language-independent identifier and with a number of information elements, called attributes, that describe the corresponding concept, and

— a set of properties, each associated with a characterization class, that are inherited over the class hierarchy, and that have also language-independent identifier and descriptive attributes (name in various languages, domain of allowed values, etc.).

Such a structure is called a **product ontology** (see 2.17), to emphasize that it is a knowledge model that defines concepts and not words. In ISO/IEC Guide 77, each particular product ontology addressing a particular product domain that is designed in compliance with the common ISO 13584/IEC 61360 dictionary model for product ontology is called a **reference dictionary** (see 2.20) for that domain, keeping in mind that it is not a dictionary of words but a dictionary of concepts.

When an attempt is made to design or to use a reference dictionary for a particular domain (e.g. fasteners), a number of issues appear, notably including the following:

— which data to use for modelling characterization classes and product properties;

— how to represent characterization class hierarchies and relationships between classes and properties;

— how to avoid redefining properties that are already defined in other reference dictionaries;

— how to avoid a combinatorial blow-up when one tries to describe all the various categories of, for example, bolts and screws;

— how to extract from a standard reference dictionary only those properties that are useful for a particular company.

To answer these questions, a common information model for product ontologies, called the **common ISO 13584/IEC 61360 dictionary model** (see 2.6), has been developed as a joint effort of ISO/TC 184, *Automation systems and integration*, SC 4, *Industrial data*, WG 2, *Standard for the neutral representation of standard parts*, and of IEC/TC 3, *Information structures, documentation and graphical symbols*, SC 3D, *Data sets for libraries*.

This model has been defined in a formal data modelling language called EXPRESS. Thus, a number of software tools have been developed that support all the concepts of this model. These tools can be used by domain experts for building their own reference dictionaries without any knowledge about EXPRESS, while ensuring that the data representation of these reference dictionaries will be exchangeable in a standard way. The only prerequisite is to understand the concepts and mechanisms defined in the model: this part of ISO/IEC Guide 77 aims to provide this background.

Due to the diversity of industrial sectors and of engineering disciplines, creating reference dictionaries that cover the whole technical domain is a huge task that can only be performed by a number of parallel groups and projects. To ensure interoperability of the developed reference dictionaries, it is crucial that the same data model be used. It is also crucial that some mechanisms be used to connect the various reference dictionaries and to reuse in each reference dictionary the relevant properties that are already defined in some other reference dictionary. For these reasons, the ISO Technical Management Board and the IEC Standards Management Board decided to establish ISO/IEC/JWG 1. Its role has been to produce a guide for specification of product properties and families on the basis of the common ISO 13584/IEC 61360 dictionary model, and to produce recommendations and guidelines on how to monitor consistency of the work items produced in accordance with the guide across all ISO/IEC Technical Committees. These guidelines are documented in this multi-part ISO/IEC Guide 77. ISO/IEC Guide 77 is not only of interest to ISO and IEC Technical Committees for developing standard reference dictionaries, it can also be useful to any group, consortium or industrial organization that needs to develop interoperable reference dictionaries.

Developing reference dictionaries is a design task. It is based on a number of design choices and decisions that need to be agreed upon in a consensual way to ensure acceptability of the developed reference dictionary. These choices are largely domain-dependent, therefore no universal and systematic methodology can be built. To help domain experts in the process, ISO/IEC Guide 77-3 provides some reports of previous experience in the use of the common ISO 13584/IEC 61360 dictionary model in reference dictionary design.

Guide for specification of product properties and classes

Part 2:
Technical principles and guidance

1 Scope

ISO/IEC Guide 77 provides general advice and guidance for the description of products and their characteristics by the use of the ISO 13584 and IEC 61360 series of standards for the creation of computer-processable reference dictionaries. This description will provide the details of the products and their properties in an unambiguous manner capable of computer communication, in a form that is independent from any proprietary application software.

NOTE 1 The term “product” is taken to include devices, processes, systems, installations, etc.

ISO/IEC Guide 77 is intended to assist the objective of enabling the flow of technical information between internal and external business partners in a cost-effective and timely manner.

The guidance given in this part of ISO/IEC Guide 77 contains technical recommendations intended to assist standardization committees and technical experts contributing their knowledge to the development of standard reference dictionaries compliant with the common ISO 13584/IEC 61360 dictionary model.

It might also be useful for information experts responsible for the exchange of technical information between business partners or for the generation of applications of ISO 13584 and IEC 61360.

This part of ISO/IEC Guide 77 is intended to support the achievement of industrial benefits of applications of the common ISO 13584/IEC 61360 dictionary model. The following are within the scope of this part of ISO/IEC Guide 77:

— general principles of product description and characterization;

— presentation of the concepts of product characterization classes, product properties, product ontology and reference dictionaries for products;

— universal identification of classes and properties;

— presentation of the modelling constructs that can be used for building reference dictionaries conforming to the common ISO 13584/IEC 61360 dictionary model;

— rules and principles for developing standard reference dictionaries;

— rules and principles for connecting standard reference dictionaries to avoid duplication and overlap;

— rules and principles for developing user-defined reference dictionaries and for connecting user-defined reference dictionaries to standard reference dictionaries;

— formats and mechanisms for exchanging reference dictionaries;

— mechanisms for connecting reference dictionaries to classification systems.

The following are outside the scope of this part of ISO/IEC Guide 77:

— an overview for ISO and IEC Technical Committees and industrial managers for the development of computer-processable product libraries, reference dictionaries and catalogues;

NOTE 2 An overview of the development of computer-processable product libraries, reference dictionaries and catalogues is provided in ISO/IEC Guide 77-1.

— case studies from experiences in the creation of reference dictionaries of product information in industrial practice;

NOTE 3 Experience gained in the creation of reference dictionaries of product information in industrial practice is provided in ISO/IEC Guide 77-3.

— categorization of products for purposes other than product characterization.

2 Terms and definitions

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

2.1
applicable property of a class
property relevant for any product belonging to a product characterization class and that can be used for characterizing it

NOTE 1 All the applicable properties of a superclass are also applicable properties for its subclasses.

NOTE 2 Only properties defined or inherited as visible and imported properties of a class can become applicable properties.

2.2
attribute
data element for the computer-sensible description of a property, a relation or a class

NOTE An attribute describes only one single detail of a property, of a class or of a relation.

EXAMPLE The name of a property, the code of a class, the measure unit in which values of a property are provided are examples of attributes.

2.3
class
class of products
abstraction of a set of similar products

NOTE 1 A product that complies with the abstraction defined by a class is called a **class member** (see 2.5).

NOTE 2 A class is an intentional concept that can take different extensional meanings in different contexts.

EXAMPLE The set of parts used by a particular enterprise and the set of all ISO-standardized parts are two examples of contexts. In these two contexts, the set of products that are considered as members of the single ball bearing class might be different, notably because employees of each enterprise ignore a number of existing single ball bearing products.

NOTE 3 Classes are structured by class inclusion relationships.

NOTE 4 A class of product is a general concept as defined in ISO 1087-1. It is therefore advisable that the rule defined in ISO 704 be used for defining the designation and definition attributes of classes of product.

NOTE 5 In the context of this part of ISO/IEC Guide 77, a class is either a product characterization class, associated with properties and usable for characterizing products, or a product categorization class, not associated with properties and not usable for characterizing products.

2.4
class inclusion relationship
relationship between classes that means inclusion of class members

NOTE 1 Therefore, if A is a superclass of A1, this means that, in any context, any member of A1 is also member of A.

EXAMPLE 1 The set of parts used by a particular enterprise and the set of all ISO-standardized parts are two examples of contexts.

EXAMPLE 2 In any context, the class capacitor includes the class electrolytic capacitor.

NOTE 2 Class inclusion defines a hierarchical structure between classes.

NOTE 3 Class inclusion is a conceptual relationship that does not prescribe anything at the data representation level; consequently, it does not prescribe any particular database schema or data model.

NOTE 4 In the common ISO 13584/IEC 61360 dictionary model, the **is-a relationship** (see 2.9) ensures class inclusion. This part of ISO/IEC Guide 77 also recommends that the **case-of** (see 2.10) relationship also ensure class inclusion.

NOTE 5 Class inclusion relationship is a case of generic relation as defined in ISO 1087-1.

NOTE 6 Class inclusion relationship is also called subsumption.

2.5
class member
product that complies with the abstraction defined by a class

2.6
common ISO 13584/IEC 61360 dictionary model
data model for product ontology, using the information modelling language EXPRESS, resulting from a joint effort between ISO/TC 184/SC 4/WG 2 and IEC/SC 3D

NOTE 1 In ISO/IEC Guide 77, a product ontology that addresses a particular product domain, based on the common ISO 13584/IEC 61360 dictionary model, is called a **reference dictionary** (see 2.20) for that domain.

NOTE 2 The current version of the common ISO 13584/IEC 61360 dictionary model recommended by ISO/IEC Guide 77 is documented in EXPRESS in both IEC 61360-5 and ISO 13584-25 (conformance classes 1,2, 3 and 4). It is also documented in UML, together with its XML exchange format, in ISO 13584-32, OntoML.

NOTE 3 The common ISO 13584/IEC 61360 dictionary model is used in a number of ISO and IEC standards, including IEC 61360-4-DB, ISO 13584-501, ISO 13399, ISO 13584-511, ISO 23584, IEC 61987-2, ISO/TS 23768.

2.7
feature
aspect of a product that can be described by a characterization class and a set of property-value pairs

NOTE In the common ISO 13584/IEC 61360 dictionary model, features and products are both represented by product characterization classes. The concept of feature is only a useful design concept to be considered when designing the class hierarchy and the associated properties of a reference dictionary.

EXAMPLE Screws can be described as consisting of different features, such as screw head, screw thread, screw shank, and screw end. Such a description, used in ISO 13584-511:2006, makes it possible to describe a huge number of different screws by a smaller number of screw classes that reference a small number of feature classes.

2.8
imported property
property defined in a class that is selected by another class of the same or of a different reference dictionary, by means of the case-of relationship, to become applicable to the latter class

NOTE 1 Only properties that are visible and/or applicable in a class can be imported from this class.

NOTE 2 Importation between classes of different reference dictionaries enables properties, e.g. properties defined for example in a standard reference dictionary, to be reused without having to be redefined.

NOTE 3 Importation between classes of the same reference dictionary acknowledges the fact that some products can perform several functions, requiring the capability to import properties from several higher level classes.

NOTE 4 When a property is imported in a new class, it keeps its original identifier; consequently, it is not necessary to duplicate all the attributes.

2.9
is-a relationship
class inclusion relationship associated with inheritance

NOTE 1 Therefore, if A1 is-a A, then each product belonging to A1 belongs to A, and all class properties described in the context of A are also automatically described in the context of A1.

NOTE 2 This mechanism is usually called inheritance.

NOTE 3 In the common ISO 13584/IEC 61360 dictionary model, the is-a relationship can only be defined between characterization classes. It is advisable that it defines a single hierarchy, and it ensures that both visible and applicable properties are inherited.

2.10
is-case-of relationship
case-of
property importation mechanism

NOTE 1 Therefore, if A1 is-case-of A, then the definition of A products also covers A1 products, thus A1 can import any property from A.

NOTE 2 The goal of the is-case-of relationship is to allow the connecting together of several class inclusion hierarchies, while ensuring that referenced hierarchies may be updated independently.

NOTE 3 There is no constraint that the is-case-of relationship define a single hierarchy.

NOTE 4 In the common ISO 13584/IEC 61360 dictionary model, the is-case-of relationship can notably be used in four cases:

a) to link a characterization class to a product characterization,

b) to import, in the context of some standardized reference dictionaries some properties already defined in other standardized reference dictionaries,

c) to connect a user reference dictionary to one or several standardized reference dictionaries, and

d) to describe a part using the properties of different classes.

When parts of class A1 fulfil two different functions, and thus are logically described by properties associated with two different classes, A and B, A1 can be connected by is-a to A, and by is-case-of to B.

2.11
leaf characterization class
leaf class
characterization class that is not further subdivided into more precise characterization classes

EXAMPLE Countersunk flat head screw with cross recess (type Y) and hexagon socket head cap screw with metric fine pitch thread are leaf characterization classes defined in ISO 13584-511.

NOTE A characterization class that is further subdivided into more precise characterization classes is called a “non-leaf characterization class” (e.g. externally-threaded component and metric threaded bolt/screw are non-leaf characterization classes defined in ISO 13584-511).

2.12
product
thing or substance produced by a natural or artificial process

NOTE In ISO/IEC Guide 77, the term "product" is taken in its broadest sense to include devices, systems and installations, as well as material, software and services.

2.13
product categorization
recursive partition of a collection of products into subsets for a specific purpose

NOTE 1 Subsets which appear in a product categorization are called **product categorization classes**, or **product categories** (see 2.14).

NOTE 2 A product categorization is not a product ontology. It cannot be used for characterizing products.

NOTE 3 No property is associated with categorizations.

NOTE 4 Several categorizations of the same set of parts are possible, depending on their target usage.

EXAMPLE The United Nations Standard Products and Services Code (UNSPSC) classification, defined by the United Nations, is an example of product categorization that has been developed for spend analysis.

NOTE 5 Using the is-case-of relationship, several product characterization class hierarchies can be connected to a categorization hierarchy to generate a single structure.

2.14
product categorization class
product category
class of products that constitutes an element of a classification

EXAMPLE Manufacturing Components and Supplies, and Industrial Optics are examples of product categories defined in the UNSPSC.

NOTE 1 No rule is given in this part of ISO/IEC Guide 77 about how to select categorization classes. This concept is introduced both to clarify its difference from characterization class, and to explain that the same characterization class can be connected to any number of categorizations.

NOTE 2 There is no property associated with a product category.

2.15
product characterization
description of a product by means of a product characterization class, to which it belongs, and a set of property value pairs

EXAMPLE Hexagon_head_bolts_ISO_4014 (product grades = A, thread type = M, length = 50, diameter = 8) is an example of product characterization.

NOTE In the above example, Hexagon_head_bolts_ISO_4014 stands for the identifier of the Hexagon head bolts product characterization class defined by ISO 4014. The names in italics between parentheses stand for the bolt properties defined in ISO 4014.

2.16
product characterization class
class of products that fulfil the same function and that share common properties

NOTE Product characterization classes can be defined at various levels of details, thus defining a class inclusion hierarchy.

EXAMPLE Metric threaded bolt/screw and hexagon head bolt are examples of product characterization classes defined in ISO 13584-511. The first characterization class is included in the second one. Transistor and bipolar power

transistor are examples of product characterization classes defined in IEC 61360-4-DB. The second one is included in the first one.

2.17
product ontology
model of product knowledge, made by a formal and consensual representation of the concepts of a product domain in terms of characterization classes, class relations and properties

NOTE 1 Product ontologies are based on a class-instance model that makes it possible to recognize and to designate the sets of products, called characterization classes, that have a similar function (e.g. ball bearing, capacitor), but also to discriminate within a class the various subsets of products, called instances, that are considered as identical. It is advisable that the rules defined in ISO 1087-1 be used for formulating designation and definitions of characterization classes. Instances have no definitions. They are designated by the class to which they belong, and a set of property-value pairs.

NOTE 2 Ontologies are not concerned with words, but with concepts, independent of any particular language.

NOTE 3 Consensual means that the conceptualization is agreed upon in some community.

NOTE 4 Formal means that it is advisable that the ontology be machine interpretable. It is advisable that some level of machine reasoning be possible over ontology, e.g. consistency checking.

NOTE 5 Identified means that each ontology characterization class and property is associated with a globally unique identifier, making it possible to reference this concept from any context.

NOTE 6 The data model for ontology recommended in this part of ISO/IEC Guide 77 is the common ISO 13584/IEC 61360 dictionary model, which is documented in ISO 13584-42 and IEC 61360-2 (conformance classes 1, 2, 3 and 4 of both documents).

NOTE 7 In ISO/IEC Guide 77, each product ontology addressing a particular product domain compliant with the common ISO 13584/IEC 61360 dictionary model is called a **reference dictionary** (see 2.20) for that domain.

EXAMPLE The reference dictionary for electric components, which is defined in IEC 61360-4-DB, is a product ontology for electric components compliant with the common ISO 13584/IEC 61360 dictionary model. A corporate reference dictionary is agreed upon by experts designated by management on behalf of the company.

2.18
property
defined parameter suitable for the description and differentiation of products

NOTE 1 A property describes one aspect of a given object.

NOTE 2 A property is defined by the totality of its associated attributes. The types and number of attributes that describe a property with high accuracy are defined in this part of ISO/IEC Guide 77.

NOTE 3 The term “property” used in this part of ISO/IEC Guide 77 and the term “data element type” used in IEC 61360 are synonyms.

NOTE 4 The term “property” used in this part of ISO/IEC Guide 77 and the term “property” used in ISO 704 are not synonyms. In ISO 704, a property is a statement about objects. In this part of ISO/IEC Guide 77, a property is used for differentiating the various products that are members of the same product characterization class. Thus, a property is associated with a domain of values, and it assigns a value in this domain for each member of a product characterization class.

2.19
property definition class
product characterization class in the context of which a product property is defined

NOTE In the common ISO 13584/IEC 61360 dictionary model, each product property has one property definition class that defines its domain of application. The property is only meaningful for this class, and all its subclasses, and it is said to be visible over this domain.

EXAMPLE In ISO 13584-511, wrenching height has nut as its property definition class, type of thread has thread as its property definition class, and major diameter of external thread has metric external thread as its property definition class.

2.20
reference dictionary
product ontology compliant with the common ISO 13584/IEC 61360 dictionary model

NOTE In ISO/IEC Guide 77, a product ontology that addresses a particular product domain, based on the common ISO 13584/IEC 61360 dictionary model, is called a "reference dictionary" for that domain.

2.21
subclass
class that is one step below another class in a class inclusion hierarchy

NOTE In the common ISO 13584/IEC 61360 dictionary model, class inclusion hierarchies are defined by the is-a relationship. They can also be established by is-case-of relationships.

2.22
superclass
class that is one step above another class in a class inclusion hierarchy

NOTE 1 In the common ISO 13584/IEC 61360 dictionary model, class inclusion hierarchies are defined by the is-a relationship. They can also be established by is-case-of relationships.

NOTE 2 In the common ISO 13584/IEC 61360 dictionary model, a class has at most one superclass specified by means of an is-a relationship.

2.23
visible property
property that has a definition meaningful in the scope of a given product characterization class, but that does not necessarily apply to the various products belonging to this class

NOTE 1 The concept of a visible property makes it possible to share the definition of a property among product characterization classes where this property does not necessarily apply.

EXAMPLE The non-threaded length property is meaningful for any class of screw, but it applies only to those screws that have a non-threaded part. It can be defined as visible at the screw level, while becoming applicable only in some subclasses.

NOTE 2 All the visible properties of a superclass that is a product characterization class are also visible properties for its subclasses.

3 General principles

3.1 The common ISO 13584/IEC 61360 dictionary model

ISO/IEC Guide 77 recommends that standard reference dictionaries be based on the common ISO 13584/IEC 61360 dictionary model. This clause outlines the origin of the common ISO 13584/IEC 61360 dictionary model. It defines the current reference documents for this model and the additional standard specifications documented either in the ISO 13584 series of standards or in the IEC 61360 series of standards.

The goal of the IEC 61360 series of standards is defined in IEC 61360-1:2004 as follows:

> "IEC 61360 provides a firm basis for the clear and unambiguous definition of characteristic properties (data element types) of all elements of electrotechnical systems from basic components to subassemblies and full systems."

The goal of the ISO 13584 series of standards is defined in ISO 13584-42:1998 as follows:

> "ISO 13584 is an International Standard for the computer-interpretable representation and exchange of parts library data. The objective is to provide a neutral mechanism capable of transferring parts library data, independent of any application that is using a parts library data system."

To achieve their goals, both standards required the definition of a model for **reference dictionary** (see 2.20) or ontology of product classes and properties. Following a joint meeting of ISO/TC 184/SC 4/WG 2 and IEC/SC 3D, it was decided to work together to develop a common model intended to be used in both standards and, more broadly, by any IEC and ISO Technical Committee.

In 1998, the first version of the model, called the common ISO/IEC dictionary model, was published. The master version was in IEC 61360-2. An informative copy was put in ISO 13584-42. This model has been further extended. Extensions include, notably, support for external files (picture, drawing) and aggregate type properties. The current version is now documented both in ISO 13584-25 and in IEC 61360-2 and IEC 61360-5. Since its publication, this model has been used by a number of ISO and IEC Technical Committees, including ISO/TC 2, ISO/TC 4, ISO/TC 29, ISO/TC 184, IEC/SC 3D and IEC/SC 65, for building standard reference dictionaries of various product domains, thus additional guidelines seemed desirable to help standardization committees in this task, which is one of the reasons why ISO/IEC Guide 77 was developed.

Besides the fact that both standards use a common model, which ensures full interoperability for the exchange of reference dictionaries between implementers of both standards, the scope of each standard is broader than just the definition of this model.

In particular, the IEC 61360 series contains also the following:

— a complete product reference dictionary for electric and electronic components, now available as an online database (see Figure 1);

— maintenance and validation procedures for standard reference dictionaries.

The ISO 13584 series contains also the following:

a) an information model for exchanging product standards, electronic catalogues or databases of components whose classes and properties are defined in accordance with the common ISO 13584/IEC 61360 dictionary model;

b) reference dictionaries for various product domains, which are published in ISO 13584 Parts 500 to 599.

Moreover, each series of standards contains some methodological guidelines for designing reference dictionaries. These guidelines complement this part of ISO/IEC Guide 77 and would prove useful for standardization committees responsible for developing new reference dictionaries.

NOTE These methodological guidelines are mainly documented in IEC 61360-1 and in ISO 13584-42.

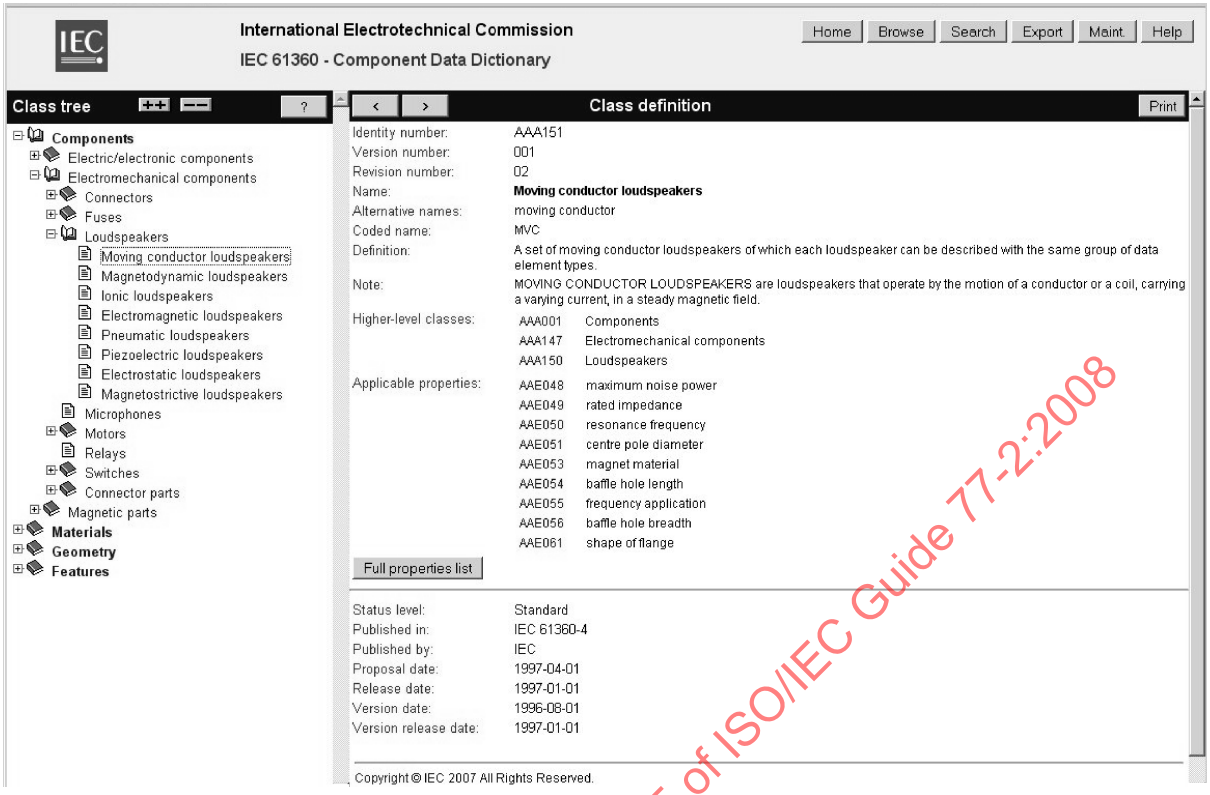

Figure 1 — The IEC 61360-4-DB online database

3.2 Simultaneous description of product characterization classes and product properties

3.2.1 General

A product may be characterized by the most specific product characterization class to which it belongs, plus a set of property-value pairs. Therefore, a reference dictionary should define both a set of product characterization classes and a set of product properties.

In the common ISO 13584/IEC 61360 dictionary model, such a definition is governed by two fundamental rules, as described in 3.2.2 and 3.2.3.

3.2.2 Rule 1: Joint definition of classes and properties

Product characterization classes and product properties should be defined together to ensure the lack of ambiguity and the consistency of both definitions.

The first rule is two-fold.

First, a product property cannot be defined without describing, in the meantime, the kind of products to which this property may be applied. Thus, each product property is associated with a product characterization class, called the **property definition class** (see 2.19), that defines its domain of application. The property is only meaningful for this class (and the classes connected to it directly or indirectly using either is-a or is-case-of). It is said to be visible over this domain. This concept of visibility is useful for distinguishing properties that seem rather similar but that are in fact different, because they are not evaluated in the same manner or in the same condition.

EXAMPLE 1 The mass of a screw and the mass of a car are not the same property, as the latter requires one to specify which particular car constituents are to be taken into account in the measuring process, e.g. car options, cooling liquid, gas.

Second, a product characterization class cannot be precisely defined without defining, in the meantime, the properties that characterize the products that are members of the class. Thus, each product characterization class is associated with those properties that are applicable to characterize its own products. Applicability of a property plays two roles:

— it provides a necessary membership criterion for class products; only those products for which each applicable property corresponds to an existing product aspect may be a member of the class;

— it may be used for characterizing class products.

Only meaningful properties may become applicable; thus, only properties that are visible for the class, or for one of its superclasses, may be applicable to a class.

When a set of classes are built using the first rule, it is often the case that the same property is needed in the definitions of several classes. To avoid a duplication of property definitions, more general classes should be defined that include the content of the various classes needing the same properties. Common properties should be defined in the context of the more general class, and the definitions of these properties are implicitly duplicated in the contexts of the classes that have an inclusion relationship with the general class.

The more general class is termed a **superclass** (see 2.22) of all the classes that it includes. The more specific classes are called **subclasses** (see 2.21) of the more general class. The mechanism which ensures that all the properties that are defined in the context of the superclass are also implicitly duplicated in the context of all its subclasses is termed inheritance. The subclass/superclass relationship that is associated with inheritance is usually called an is-a relationship.

The choice of the superclasses to be defined should represent new abstractions that correspond, as much as possible, to kinds of products that are familiar in the target product domain.

EXAMPLE 2 According to the phase of a manufacturing process, an engineer might want to characterize a bearing, a ball bearing or a thrust angular contact ball bearing. Such kinds of products are examples of abstractions that can be defined as classes.

3.2.3 Rule 2: Single is-a hierarchy with inheritance

Product characterization classes corresponding to the same kind of products should define a single product characterization class hierarchy where:

— **the content of each subclass is included in the content of its superclass, and**

— **properties applicable to a superclass are also applicable to all of its subclasses.**

In some cases, several property factorizations are simultaneously possible, and thus several superclasses could be built for the same class, each superclass factorizing a different set of properties. To ensure that the class structure remains both understandable and manageable, a second rule specifies that each class should have at most one superclass in the is-a hierarchy.

NOTE 1 When several property factorizations are simultaneously possible, various criteria can be used for selecting one particular hierarchy. These criteria include factorizing the maximal number of properties and defining superclasses that are more understandable.

NOTE 2 When parts of a class A1 fulfil two different functions and are thus logically described by properties associated with two different superclasses, A and B, one particular superclass, e.g. class A, needs to be selected for an is-a relationship with A1. Then class A1 can be connected to class B using the is-case-of relationship (see 3.5.2). Through the is-case-of relationship, class A1 can import explicitly the needed properties from B. But there is no inheritance from B to A1: not all of class B's properties implicitly exist in the context of A1, and when new properties are added to B, or when some of B's properties are modified, A1's properties do not automatically change.

Thus, product characterization classes may be defined at various levels of details, each superclass being a generalization of a set of more specialized classes.

3.2.4 Structure of a reference dictionary

Therefore, for each particular technical domain, a reference dictionary consists of two parts:

a) a class tree, or a set of class trees, where product classes and properties are identified and connected together (see the left part of Figure 2), and where properties defined at each level are inherited by the lower levels (is-a relationship), and

b) a set of attributes that describes successively each product characterization class and each product property (see the right part of Figure 2).

Advantages of this approach include those listed below.

— The tree structure allows one to organize and to manage all the properties of a product domain; it makes them retrievable and manageable considering their great numbers.

— Inheritance of properties makes it possible both to define each property precisely, i.e. in the context of a class that defines its domain of application (see Rule 1 above, in 3.2.2), and to avoid duplication, i.e. to avoid redefining a similar property for each subclass. A property is defined precisely in the context of the more general product characterization class, where it is comprehensible, and it may then be reused unambiguously in all its subclasses.

— Products may be characterized at various levels of details, e.g. bearing, ball bearing, and thrust angular contact ball bearing.

— In a product database, each product needs only to be recorded in one product characterization class. A query over products of some product characterization class is automatically computed by performing the same query over all the subclasses of this class.

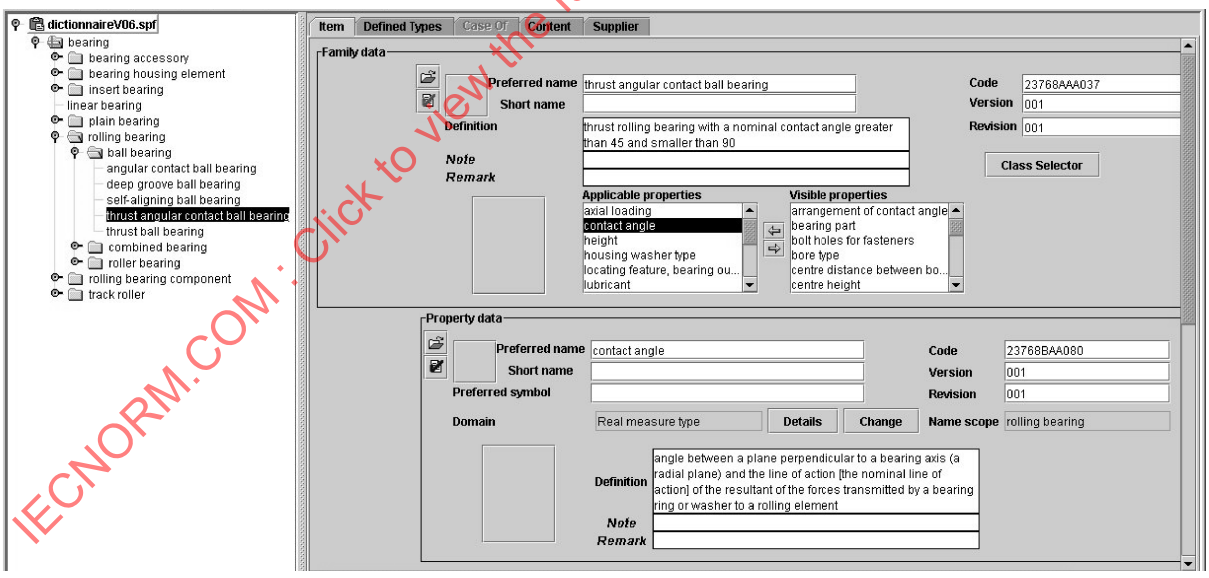

Figure 2 — Structure of a reference dictionary

Nevertheless, two kinds of difficulties should be noticed.

In the first place, setting up a product characterization class hierarchy is a design choice. Therefore, it is often necessary to iterate when new properties are identified during the design process of such a hierarchy.

Secondly, in order to address complex reality, some additional mechanisms need to be defined that will make the model less simple. These mechanisms are described in Clauses 5 and 6.

3.3 Attribute and relationship-based definitions

Identifying without ambiguity the various concepts of a reference dictionary, i.e. product characterization classes and product properties, cannot be done by single phrases. Following the approach known as information modelling, each concept is represented by:

a) attributes, i.e. pieces of information that define one aspect of the concept (e.g. name, textual definition, picture, domain of values of a property), and

b) relationships between concepts (e.g. the superclass, if any, of a product characterization class; the properties that are applicable to products belonging to a product characterization class).

NOTE In the common ISO 13584/IEC 61360 dictionary model, these relationships are also modelled by attributes that contain references to related concepts.

All these attributes are described in Clauses 4, 5 and 6. They are explicitly represented in the common ISO 13584/IEC 61360 dictionary model and can be set by the various software tools capable of designing reference dictionaries.

Both classes and properties are concepts as defined in ISO 1087-1. Thus, the methodology and guidelines defined in ISO 704 to formulate designations and definitions of concepts on the basis of a concept system should be used for assigning values to the designation and definition attributes of classes and properties. The class and property structure defined in accordance with 3.2 constitutes the concept system upon which definitions should be based.

The following class and properties attributes record terminological data that should be formulated in accordance with ISO 704:

— preferred name;

— synonymous name;

— short name;

— definition;

— note;

— preferred symbol;

— synonymous symbol.

3.4 Universal identification of classes and properties

Once a concept is defined in a reference dictionary, this concept should be usable in any business context, e.g. product database, electronic commerce, concurrent engineering.

In the common ISO 13584/IEC 61360 dictionary model, this is made possible by assigning automatically to each product characterization class and to each product property an identifier that is unique over all the reference dictionaries, regardless of who developed them.

In accordance with ISO/IEC 11179-5, this identifier, called international registration data identifier (IRDI), consists of three parts:

— registration authority identifier (RAI);

— data identifier (DI);

— version identifier (VI).

NOTE 1 ISO/IEC 11179-5 only specifies the data elements of an IRDI, but not the syntax. The syntax of an IRDI defined in this clause is the one that needs to be used with the common ISO 13584/IEC 61360 dictionary model. Thus, the rules below refine the rules for IRDIs given in ISO/IEC 11179-5.

The RAI is a universal identifier of the source organization for the specification of the concept, class or property. The RAI should be assigned in accordance with ISO 13584-26, which refines ISO/IEC 6523-1.

NOTE 2 An RAI consists of:

a) the international code designator (ICD) of any organization identification scheme conforming to ISO/IEC 6523-1 (e.g. SIREN numbers, Deutscher Industrie- und Handelskammertag Scheme, NATO ISO 6523 ICDE coding scheme), and

b) the organization identifier assigned to the source organization in this coding scheme.

This is optionally followed either by the organization part identifier (OPI) and the OPI source indicator (OPIS) or, for concepts defined by International Standards, by the number of this International Standard. See ISO 13584-26 for details.

The DI needs only to be unique within the whole set of concepts defined by the same source organization. Moreover, the common ISO 13584/IEC 61360 dictionary model restricts the total length of the DI and the use of three characters: "." (dot), " " (space), and "-" (hyphen). See ISO 13584-42 or IEC 61360-2 for details. This part of ISO/IEC Guide 77 also recommends avoiding the use of "#" (hash) and "*" (star), as they are reserved for the latter standardization.

NOTE 3 Besides being unique over all the concepts defined by the same source organization, the data identifier of a property, as defined in the common ISO 13584/IEC 61360 dictionary model, consists of three parts:

a) the data identifier of the class in the context of which the property is defined, i.e. its definition class,

b) a dot (".") separator, and

c) a property code.

The VI, a positive integer, should be assigned in increasing order for the various versions of the same concept.

Figure 3 shows the universal identification of a product property defined in IEC 61360-4-DB.

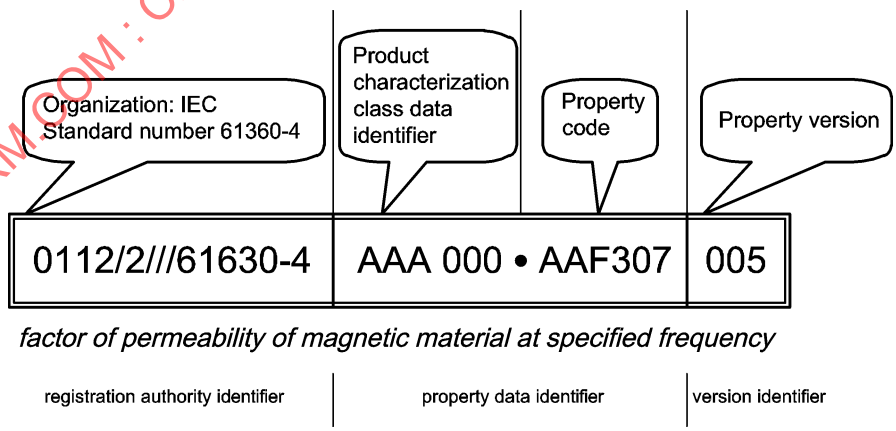

Figure 3 — International registration data identifier (IRDI) of a property defined in IEC 61360-4-DB

3.5 Modularity

3.5.1 Avoiding duplication across reference dictionaries

One expected result of ISO/IEC Guide 77 is that a number of reference dictionaries addressing various product domains will be developed in such a way that they may be used together seamlessly, but if all these reference dictionaries are developed independently of each other, a number of overlaps would occur. Indeed, several properties and classes might prove useful in a number of different product domains. If such properties and classes are modelled several times, they would be associated with different identifiers. Thus, the same pieces of information would be described differently according to the reference dictionaries they reference.

To provide for the parallel development of reference dictionaries corresponding to various product domains, while avoiding duplication of work, interfacing mechanisms are defined. If these mechanisms are used, a reference dictionary, as defined in this part of ISO/IEC Guide 77, will be a modular structure that is self-contained and of manageable size. The interfacing mechanisms defined in this clause allow one both to reuse in the context of one particular reference dictionary properties and classes defined in the context of another one, and to reference simultaneously several reference dictionaries as if they were integrated into a single reference dictionary.

NOTE The mechanisms defined in this clause assume that possible copyright issues for referencing a property or a class of a reference dictionary from another reference dictionary have been solved, and that the level of consensus on the reference dictionary from which a property or a class is borrowed is similar to the level of consensus on the reference dictionary where the property or class is imported. This is notably the case when both dictionaries are ISO or IEC standards. In other cases, creating a new property or class might be considered.

3.5.2 Sharing properties among reference dictionaries

A product characterization class provides a context in which a given property may be described in a clear and unambiguous manner by making reference to the kind of products to which the property applies. However, a property defined in a class C1 of one particular reference dictionary might prove useful in a class C2 of some other reference dictionary.

To allow for this, the common ISO 13584/IEC 61360 dictionary model introduces a particular articulation mechanism between reference dictionaries, called is-case-of. If C2 declares that it is-case-of C1, this means that the C1 definition also covers C2's products. Thus, C2 may import, through this declaration, any property defined for C1. Imported properties preserve their unique identifier and all their other attributes, and they may be used in the context of C2, and of all its subclasses. Thanks to the unique identifier, the property remains understandable even for those systems that only know C1.

In fact, there are three cases where the definitions given in class C1 also cover C2's products.

a) The first case is when the products that are characterized as being members of C2 are also members of C1: in any context, the extent of C2 is included in the extent of C1. In this case, the is-case-of relationship establishes a class inclusion relationship.

   EXAMPLE 1 Let us assume that the MFP Company produces a set of stainless steel hexagon bolts of various sizes which have a metric ISO thread. If the MFP Company characterizes these products as being members of an MFC stainless steel hexagon bolt class, it is clear that members of this class are also members of the metric threaded bolt/screw standard characterization class, as defined in ISO 13584-511:2006.

b) The second case is when the products that are characterized as being members of C2 embed the capabilities provided by C1's products. In this case, the extent of C2 may be considered as included in the extent of C1. In this case, the is-case-of relationship establishes a class inclusion relationship.

   EXAMPLE 2 Most current mobile phones also have camera capability, therefore such mobile phones can also be considered to be digital cameras. Thus, camera properties such as image max resolution or effective pixels apply to these products.

c) A third case is when one wants to expose, at the level of a product considered as a whole, a property which is a property of a part of this product.

EXAMPLE 3 The horsepower of a car can be useful for characterizing cars. In fact, the horsepower of a car is the horsepower of the engine of the car. If a standard ontology exists for an engine that has already defined the horsepower property, it would be useful to be able to use this property for characterizing a car.

The structure of the ISO 13584/IEC 61360 model makes the three cases described above in a), b) and c) possible. Nevertheless, using is-case-of for expressing product composition, as is done in case c), may lead to ambiguities when the composite product includes several similar components.

EXAMPLE 4 Let us assume that the car to be characterized in EXAMPLE 3 is a hybrid car, with both a fuel engine and an electric engine. The horsepower property becomes ambiguous when imported within the context of this car: does it refer to the horsepower of the fuel engine, or the horsepower of the electric engine?

For this reason, this part of ISO/IEC Guide 77 discourages the use of the is-case-of relationship for importing in the context of a composite product a property of one of its components. This should be done by making explicit the composition structure, as explained in 4.5.5.

When a particular class of components may be considered as embedding the capabilities of two different classes, as in case b) above, it is a design choice whether this class of components is included into both classes (in which case the is-a and is-case-of relationships should be used), or whether it is a composite product that belongs to one class and that embeds a component belonging to the other class.

EXAMPLE 5 A class of mobile phone with camera capability can be defined either as a subclass of mobile phone (is-a) that is-case-of the digital camera class, or as subclass of mobile phone that embeds, though a camera property, a component that belongs to the digital camera class.

Note that the case-of mechanism may also be used, within the same reference dictionary, when a product characterization class gathers properties from two higher-level classes. Due to the single is-a hierarchy with inheritance rule, this class is to be defined as an is-a subclass of either. By a case-of declaration, it imports from the other class those properties that are needed in its own context.

3.5.3 Sharing classes among reference dictionaries

Thanks to the universal identification of classes and properties, when needed, a class defined in a reference dictionary may be directly used in the context of another reference dictionary if the same resource proves also useful in the latter context.

For example, ISO 13584-511 defines the product characterization class of thread with metric external thread and metric internal thread as particular subtypes (see Figure 4). For any class that contains a threaded feature, this threaded feature may be represented by a property, whose domain of values is the thread characterization class defined in ISO 13584-511.

3.5.4 Importation of non-technical properties

Most reference dictionaries need properties such as part identifier and manufacturer name. These properties are non-technical and should not be created repeatedly in each domain-specific reference dictionary. Wherever appropriate, all the non-technical properties should be imported from already existing dictionaries using the is-case-of relationship.

Figure 5 presents the planned structure resulting from the use of this part of ISO/IEC Guide 77 by the various standardization committees. The following points are noteworthy:

— a number of domain-specific reference dictionaries are developed by the various committees or joint groups;

— each group avoids overlap when appropriate by using either the case-of relationship (for property reuse) or class reference (for class reuse) with other domain-specific reference dictionaries;

— all the domain-specific reference dictionaries refer to some common high-level reference dictionary (or reference dictionaries) for importing non-domain-specific properties.

From this structure, any organization may build its own global reference dictionary while preserving interoperability with standard reference dictionaries developed following this part of ISO/IEC Guide 77.

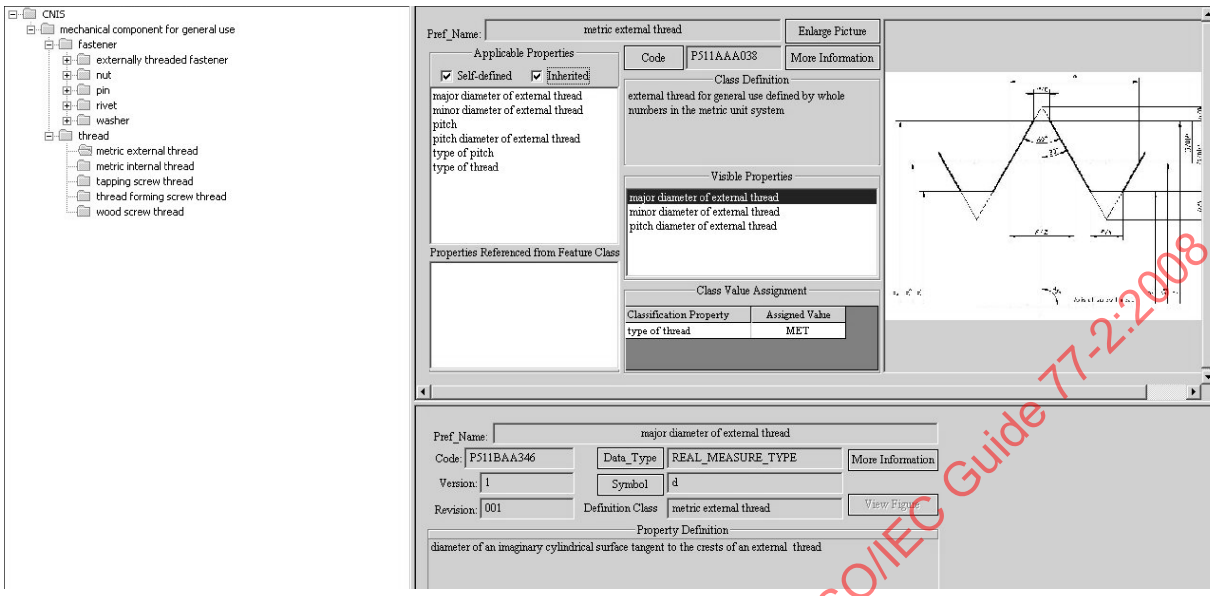

Figure 4 — Thread characterization classes in ISO 13584-511

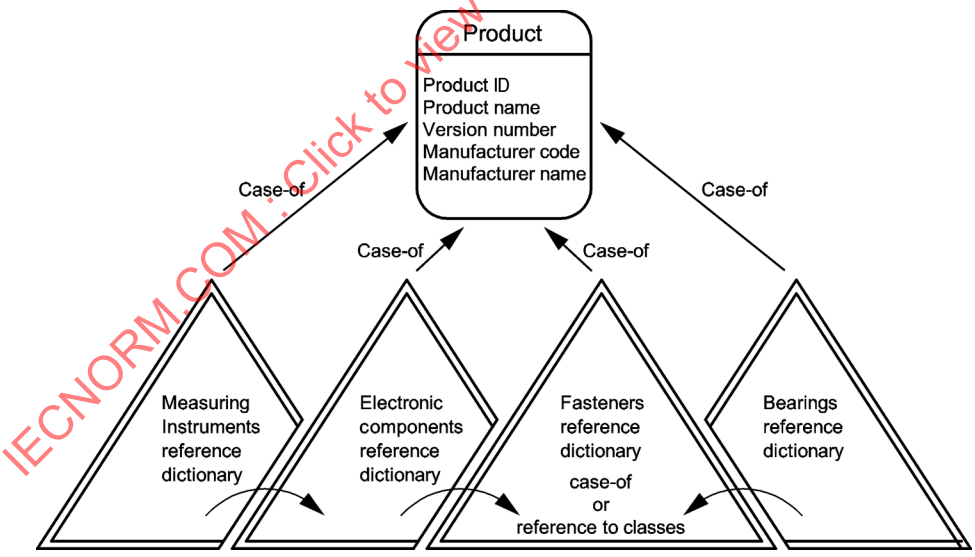

Figure 5 — Target structure of standard reference dictionaries

4 Specifying properties

4.1 Properties to be defined

The properties that should be specified in a standard reference dictionary include those listed below.

a) All those properties that can be useful when searching products from any class of the characterization class hierarchy should be defined in the context of this class.

b) Properties whose values are standardized (product standards) and/or whose names are defined by international terminology standards within the target product domain should be defined in the context of the appropriate product characterization class.

c) Other properties may be added as appropriate.

Conversely, properties that should not be specified in a standard reference dictionary include those listed below.

— Properties that already exist in another standard reference dictionary compliant with the common ISO 13584/IEC 61360 dictionary model should be, as far as possible, imported using the case-of relationship.

— Non-technical properties (e.g. price, part number) should not be redefined in each particular reference dictionary, but should be imported whenever possible.

4.2 Properties and associated attributes

Any property is described by a number of specified attributes, some of them being mandatory and others being optional. Attributes are also used for representing relationships between properties, or between properties and classes, by means of references to the other entity. This is the case, for example, for the definition class, which associates to a product property the product characterization class in the context of which the product property is defined. All attributes are intended for the unambiguous identification, definition and management of properties and to facilitate their handling and maintenance.

Table 1 defines these attributes and specifies whether they are mandatory (“Mand”) or optional, and whether their content may be translated (“Trans”) or not.

All properties within a reference dictionary should be drafted according to a consistent scheme. Therefore, each particular reference dictionary should specify which optional attributes are used and not used in its content. Definitions should be drafted in accordance with ISO 704 and ISO/IEC 11179-4.

The code and version are part of the IRDI, which provides a globally unambiguous, language-independent means to refer to a property. The code should not contain any inherent meaning.

The definition class specifies the set of products for which the property may be used by means of a reference to a class.

Table 1 — Property attributes

Attribute	Mand[a]	Trans[b]	Meaning
Code	yes	no	An alphanumeric set of characters without dot and hyphen, unique within the property definition class
Version	yes	no	Natural number. A new version is issued whenever a change in an attribute influences the use of the property (e.g. a broader value domain)
Definition class	yes	no	Reference to the product characterization class in the context of which the property is defined
Revision	yes	no	Natural number. A new revision is issued whenever a change in an attribute does not influence the use of the property (e.g. typographical errors)
Date of original definition	yes	no	Date where the property was first introduced in the reference dictionary
Date of current version	yes	no	Date on which the current version of the property was defined
Date of current revision	yes	no	Date on which the current revision of the property was defined
Preferred name	yes	yes	Single- or multi-words label, for human-readable designation of the property
Synonymous name	no	yes	Single- or multi-words label. Alternative designation that represents the same property
Short name	no	yes	Single- or multi-words label. A meaningful abbreviation of the preferred name. If standardized abbreviations exist, they should be used
Definition	yes	yes	An unlimited text that describes textually the meaning of the property
Source document of definition	no	yes	Reference that identifies the document, or the various language-specific documents, from which the definition of the property has been derived, if any
Note	no	yes	Explanatory text to further clarify the definition of the property
Remark	no	yes	Explanatory text to further clarify the meaning of the usage of the property
Preferred symbol	no	no	A symbol that represents the property
Synonymous symbol	no	no	Up to two alternate symbols that represent the property
Figure	no	no	Illustration to clarify the meaning of the definition of a property
Property type classification	no	no	Reference to category of properties to which the property belongs
Domain	yes	no	The domain of values that may be assigned to the property (see 4.3)
Formula	no	no	Rule or statement in mathematical form expressing the semantics of a quantitative property
Depends on	no	no	For those properties that are context dependent properties (see 4.4) references to the context parameters on which they depend
Value format	no	no	Type and length of the recommended presentation for displaying the value of a property
Unit of measure	yes/no	no	Prescription of the unit in which the value of a quantitative property is expressed (mandatory for quantitative property)
Alternative unit	no	no	Prescription of the possible alternative units in which the value of a quantitative property may also be expressed

[a] "Mand": is the attribute mandatory or not?

[b] "Trans": can the content of the attribute be translated or not?

4.3 Properties and associated values

The domain of allowed values of a property is defined by an entity called a data type. This data type is specified by an attribute when the property is defined. The common ISO 13584/IEC 61360 dictionary model defines a number of data types presented in Table 2.

Properties for which the data type is class instance type takes, as a value, a product characterized as a member of a product characterization class. For instance, a property whose domain is the thread class, defined in ISO 13584-511, makes it possible to model a threaded element of a particular product. This threaded element is itself defined by property values (see 4.5.5). Hence, the class instance type data type provides for feature-based modelling and for assembly modelling (see 4.5.5 and 4.6).

Table 2 — Data types

Type Name	Meaning
level type	an association of the values of a property that is a physical quantity with one or more of the following keywords: min: the minimal value of the physical quantity nom: the nominal value of the physical quantity type: the typical value of the physical quantity max: the maximal value of the physical quantity
class instance type	an instance of a reference dictionary product characterization class
Boolean type	either true or false
number type	an integer or real number
string type	a sequence of characters
int type	an integer
int currency type	an amount of a currency whose quantity is given as an integer
int measure type	a dimensioned integer. The dimensional units are given in the data type definition
non-quantitative code type	the selection of a code from an enumerated set of codes. Each code has an associated meaning
non-quantitative int type	the selection of an integer from an enumerated set of integers. Each integer has an associated meaning
real type	a dimensionless real number
real currency type	an amount of a currency whose quantity is given as a real number
real measure type	a dimensioned real number. The dimensional units are given in the data type definition
external item type	an external file that constitutes the value of the property
array type	an ordered collection of values, of fixed length, whose members are indexed by a range of consecutive integers
bag type	an unordered collection of values that may contain duplicates
list type	an ordered collection of values
set type	an unordered collection of values that does not contain duplicates
named type	any type previously defined and associated with an identifier

Non-quantitative code type allows the definition of a property that identifies or describes a product by means of codes, abbreviations, names, references or descriptions. Each value of the domain of values of a non-quantitative code type consists of a language-independent code (of up to 18 alphanumeric characters), and of a definition documented by a preferred name, possibly a short name, an icon and a possible source document where the value was defined. Figure 6 shows an example of non-quantitative code type together with user-interfaces describing it in a reference dictionary editor (right side) and displaying it in a reference dictionary browser (left side).

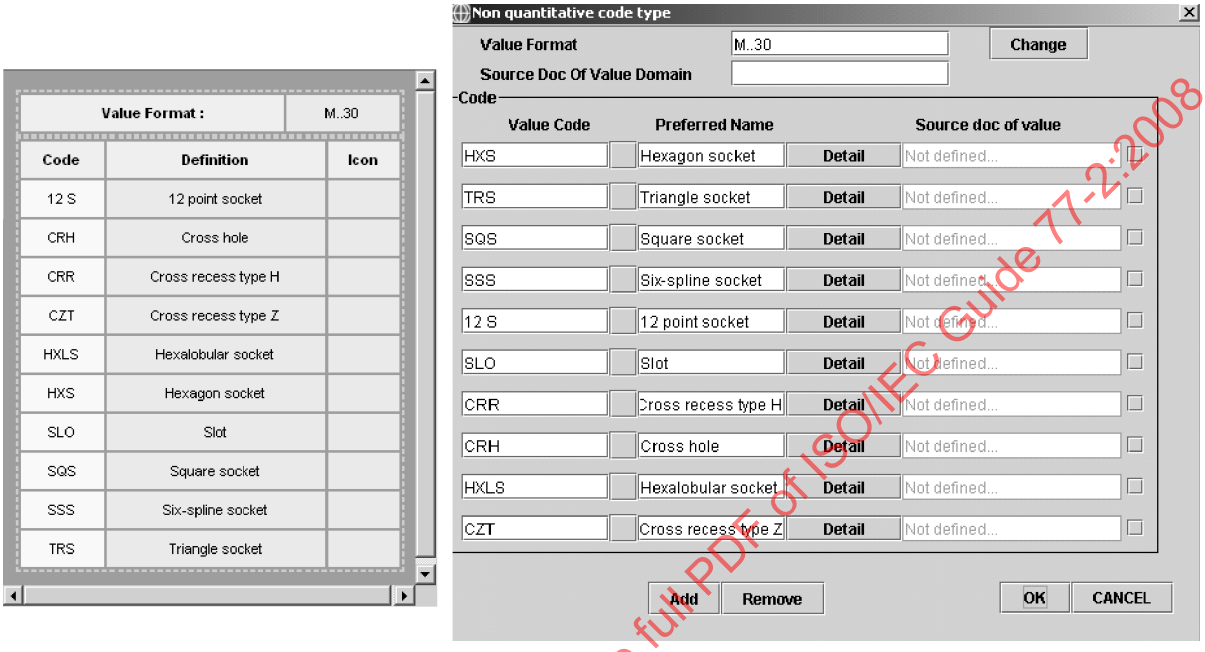

Figure 6 — Example of non-quantitative code type

External item type allows the definition of a property of whose value is a link to a file. This data type provides for describing class instances, e.g. by a data sheet, a drawing, a maintenance booklet, etc.

Array type, bag type, list type and set type are called aggregate types. The value of a property of one of these types is a collection of one or more values of some underlying type. The elements of an aggregate value may be values of any type shown in Table 2, including another aggregate. All elements of an aggregate should be of the same data type.

To avoid redefining repetitively a complex domain structure, it is possible to associate an international registration data identifier to any type of domain of values. Such a named type may then be referenced by several properties. Named types are defined in the context of a product characterization class and they may only be used in this class, its subclass and its case-of class.

4.4 Properties and context of evaluation

No object in the real world can be considered to be isolated from its environment. Quantitative properties that are measured, therefore, need to be related to conditions under which their values were obtained.

NOTE 1 Strictly speaking, it is advisable that any measure of a dimension be accompanied by quoting the temperature at which the measurement was made, even though in practice this is usually not necessary. However, for example, the resistance of an electric thermistor can strongly depend on ambient temperature, hence it is advisable that this information always be supplied.

When the evaluation context needs to be made explicit, two approaches are possible, as outlined below.

a) If it is decided that the value of a particular property should always be evaluated in a given context, this context should be described in the definition of the property.

b) Alternatively, the evaluation context should be described by means of context properties (called context parameters) that are referenced by the context dependent property. This reference is done by means of a particular attribute of context dependent properties.

Consequently, three categories of properties exist:

— context independent properties, whose context of measure either does not exist, or is described in the definition of the property,

— context dependent properties, whose context of measure is explicitly represented by means of a reference to some context parameters, and

— context parameters that represent the various physical quantities able to characterize a measure context (e.g. environmental temperature) or the context in which a product is inserted.

The category to which a property belongs needs to be defined when the property is created.

NOTE 2 Context parameters are called “condition data element types” in IEC 61360.

4.5 Product assemblies and their properties

4.5.1 General

4.5 and 4.6 provide additional guidelines for those reference dictionaries where some explicit modelling of product structure is needed. They may be omitted in a first reading of this part of ISO/IEC Guide 77.

4.5.2 Representing assemblies

Any kind of product, whether it is a piece part (e.g. a resistor or a screw) or an assembly of other products (e.g. a power transformer or a measuring instrument), may be directly characterized by a class to which the product belongs and a set of property-value pairs.

When it is necessary not only to model the global product but also to model the components from which the assembly is made, two other approaches are possible:

a) characterize each component both by a product characterization class and by properties;

b) characterize each component as above, and characterize the assembled product itself by a product characterization class and by properties that may be either assembly properties or references to its constituent components.

Thus, an assembly class may be defined in a reference dictionary by three approaches described in 4.5.3 to 4.5.5. All these approaches are permitted by the IEC 61360 and ISO 13584 series of standards, and it is a design choice as to which approach should be followed in each particular situation.

4.5.3 Characterization-by-properties

In the characterization-by-properties approach, only the global properties of the assembly are defined in the reference dictionary.

EXAMPLE Using the characterization-by-properties approach, a bolt-nut-washer assembly product can be characterized as shown in Table 3.

Table 3 — Example of characterization-by-properties

product characterization class	property-value pairs	
bolt-assembly	nominal diameter	5
	maximum assembly width	25
	kind of screw	“hexagonal”
	kind of nut	“hexagonal”
	kind of washer under head	“flat washer”
	kind of washer under nut	“spring washer”

4.5.4 Characterization-by-bill-of-material

In the characterization-by-bill-of-material approach, the assembly is only described by its constituent components (bill of material).

EXAMPLE Using the characterization-by-bill-of-material approach, a bolt-nut-washer assembly product can be characterized as shown in Table 4.

Table 4 — Example of characterization-by-bill-of-material

product characterization class	property-value pairs	
hexagon bolt	nominal diameter	5
	total length	30
hexagon nut	nominal diameter	5
	distance across flat	12
	width	3,5
flat washer	internal diameter	5
spring washer	internal diameter	5
	height	0,5

4.5.5 Characterization-by-structure

In the characterization-by-structure approach, the assembly is described not only by its constituent components, but also by

a) a product characterization class that represents the assembled product itself,

b) global assembly properties, and

c) explicit structure description.

The assembly structure is defined by means of properties that reference its constituent components.

EXAMPLE Using the characterization-by-structure approach, a bolt-nut-washer assembly product can be characterized as shown in Table 5.

NOTE When the characterization-by-structure approach is used for characterizing a particular class of products in a reference dictionary, any of the three kinds of representations listed below can be used for characterizing products of this class of products or for representing them at exchange time.

— The constituent components can be represented as embedded within the assembly representation. This means that constituent component data belong to the assembly data and have the same life cycle.

— The constituent components can be represented outside the assembly representation and are referenced from the assembly. This means that each constituent component and the assembly itself has its own life cycle.

— The constituent components can be not explicitly represented. The constituent component properties are represented directly as properties of the assembly product instance using the path, also called the dot notation, that defines the meaning of the attribute (e.g. the bolt nominal diameter = 5; the nut width = 3,5). This assembly representation, called representation-by-flattened-structure, greatly simplifies representation of simple assemblies. It is in particular very useful when a component is defined in a reference dictionary using a feature-based approach (see 4.6).

Table 5 — Example of characterization-by-structure

product characterization class	property-value pairs		property-value pair of sub-objects	
bolt assembly	maximum assembly width	25	—	
	maximum strength	10 000	—	
	the bolt	hexagon bolt	nominal diameter total length	5 30
	the nut	hexagon nut	nominal diameter width	5 3,5
	washer under head	flat washer	internal diameter	5
	washer under nut	spring washer	internal diameter height	5 0,5

4.6 Properties and features

Some single components may consist of different aspects, or features. If each feature may belong to various classes, each class being characterized by different properties, there are a huge number of different classes of components characterized by different sets of properties. In this case, each class of features may be defined as a characterization class, or a characterization class hierarchy. The product itself is described by properties whose domains of values are defined by these feature classes.

NOTE In the common ISO 13584/IEC 61360 dictionary model, features and products are represented by the same characterization class structure. The concept of feature is only a useful design concept to be considered when designing the class hierarchy and the associated properties of a reference dictionary.

EXAMPLE 1 Screws consist of different features, such as screw head, screw thread, screw shank, screw end. Each screw feature belongs to various classes, such as, for screw head: hexagon head, cheese head, and raised countersunk head. Each feature class is characterized by different properties, such as width across flats for a hexagon head, diameter of head for a cylindrical head, and radius of the raised portion of the head for a raised countersunk head. Therefore, there are a huge number of different classes of screws characterized by different sets of properties. If it is needed to model all these properties, it is advisable to use the characterization-by-structure approach:

— each feature will be described by a product characterization class associated with particular properties;

— the screw itself will contain, besides some global properties (e.g. material identification), one property whose value is the screw head, another property whose value is the screw thread, etc.

EXAMPLE 2 The standard reference dictionary for fasteners, documented in ISO 13584-511 and presented in Figures 4 and 8, contains an example of such a characterization-by-structure approach using features. An externally-threaded fastener is described as an assembly characterized-by-structure of five features: the end, the head, the internal driving, the shank and the thread. Figure 7 shows the properties applicable to a Hexalobular socket characterization class, which is a subclass of the internal driving feature class.

5 Designing product characterization class hierarchies

5.1 Product characterization classes to be defined

Each standard product characterization class hierarchy should be able to cover at least all the kinds of products that are either standardized (product standards) or described (terminology standards) by international standards within the target product domain.

A category of products that is neither standardized nor described explicitly by International Standards of the target product domain may be represented as a product characterization class and added in the product characterization class hierarchy, as appropriate. The class needs to be approved in accordance with the same procedure as those classes that are either standardized or described explicitly by International Standards.

The properties defined in standard product characterization reference hierarchies should also be usable to define non-standard products.

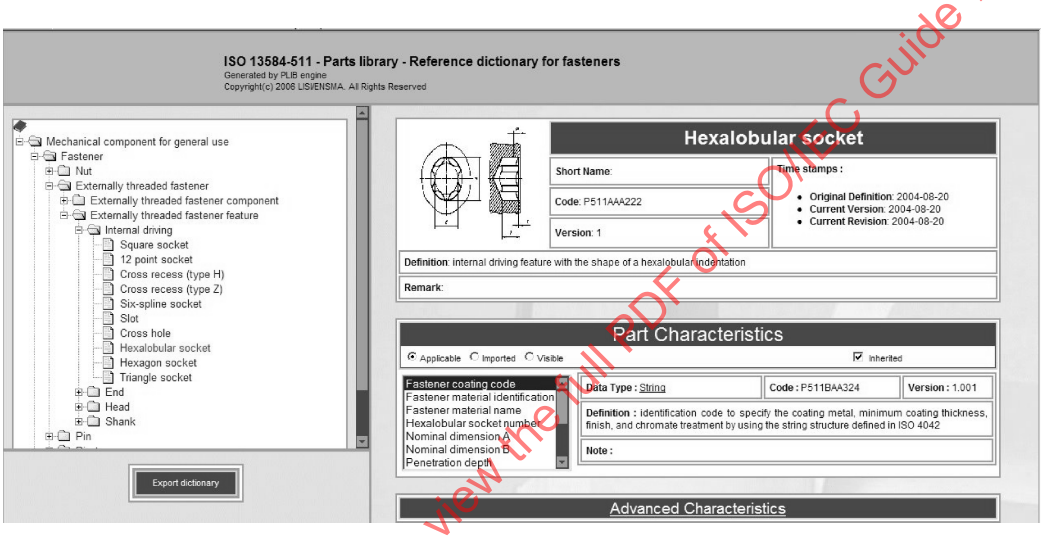

Figure 7 — ISO 13584-511 use of feature-based modelling

At the lower level of the product characterization class hierarchy, each subset of products belonging to the target product domain for which items a) and b) (below) apply should be defined as a leaf class of the product characterization class hierarchy:

a) all products share the same set of characteristic properties, and

b) the set of characteristic properties is different from that of any other subset.

Above the leaf classes, superclasses should be created when

— the corresponding concepts are of widespread use in the corresponding domain, and/or

— these superclasses prove useful to factorize properties across several lower level classes.

Each particular hierarchy, developed by a particular group, should be rooted by a class node that includes all the products intended to be addressed by the product characterization class hierarchy. The definition of the root class should be drafted accordingly. If the target domain consists of the union of several sub-domains and seems too broad to be managed conveniently, the reference dictionary may be split into several reference hierarchies, as appropriate. This splitting should notably be considered when various parts of the hierarchy should be classified in different categories of the International Classification for Standards (ICS).

A hierarchy is best set up in several steps. Once a first hierarchy has been set up, properties may be assigned to classes. This is an iterative process, which in practice should be run repetitively, since the definition of classes and those of properties are correlated with each other (see 5.3).

As a rule, hierarchies should not be deep. A characterization class should never be created when the same content may be defined just by restricting the values of some properties of an existing class of the same reference dictionary. This ensures the uniqueness of the characterization of any product.

5.2 Product characterization classes and associated attributes

Any product characterization class is described by a number of attributes specified in the common ISO 13584/IEC 61360 dictionary model, some of them being mandatory and others being optional. Attributes are also used for representing relationships between entities by means of references to these entities, e.g. the optional superclass attribute establishes the is-a relationship by associating a class with its superclass.

Table 6 defines these attributes and specifies whether they are mandatory ("Mand") or optional, and whether their content may be translated ("Trans"), or not.

5.3 Connecting classes and properties: visibility and applicability of properties

5.3.1 Product characterization class hierarchies are tree structured with inheritance of applicable properties. Unfortunately, it is often not possible to find a tree structure of product characterization classes such that when a property is defined in some class,

— it applies to all its subclasses, and

— it does not apply to other parts of the hierarchy.

5.3.2 Nevertheless, ISO 13584/IEC 61360 requires that

a) each property be defined only once and be defined in the context of one product characterization class where it is comprehensible, and

b) for each product characterization class, it is explicitly specified which property applies.

This requirement is addressed in the following way.

— Each property is defined in the highest class in which it may be defined in a comprehensible way, even if it does not apply to some products of this class. This is the definition class of the property. By inheritance, this property is also defined in all the direct or indirect subclasses of the definition class. It is said that the property is visible in these classes.

EXAMPLE 1 The property single part mass might be defined in a comprehensible way at the general level of a single part product, even though it does not apply to some single part products, e.g. software products. For instance, the definition might be "mass of a product that consists of a single part in standard conditions when the product is a material product".

— This property may then be defined as applicable in any product characterization class where it is visible or in any product characterization class which is case-of a class where it is visible. By inheritance, it becomes applicable in all the subclasses of this product characterization class.

EXAMPLE 2 In EXAMPLE 1 above, single part mass can be specified as applicable to the class externally threaded fastener component. By inheritance, it becomes applicable to any particular subclass of externally threaded fastener component, such as set screw.

Table 6 — Class attributes

Attribute	Mand[a]	Trans[b]	Meaning
Code	yes	no	Alphanumeric set of characters without dot and hyphen, unique within the set of classes defined by the same organization or the same standard
Version	yes	no	Natural number. A new version is issued whenever a change in an attribute influences the use of the product characterization class (e.g. a broader value domain)
Information supplier	yes	no	Reference to the organization which defines the product characterization class
Revision	yes	no	Natural number. A new revision is issued whenever a change in an attribute does not influence the use of the product characterization class (e.g. typographical errors)
Date of original definition	yes	no	Date on which the product characterization class was first introduced in the reference dictionary
Date of current version	yes	no	Date on which the current version of the product characterization class was defined
Date of current revision	yes	no	Date on which the current revision of the product characterization class was defined
Preferred name	yes	yes	Single- or multi-words label, for human-readable designation of the product characterization class
Synonymous name	no	yes	Single- or multi-words label. Alternative designation that represents the same product characterization class. None, one or two synonymous names per language are allowed
Short name	no	yes	Single- or multi-words label. A meaningful abbreviation of the preferred name. If standardized abbreviations exist, they should be used
Definition	yes	yes	Unstructured text string that describes textually the meaning of the product characterization class
Source document of definition	no	yes	Reference that identifies the document, or the various language-specific documents, from which the definition of the product characterization class has been derived, if any
Note	no	yes	Explanatory text to further clarify the definition of the product characterization class
Remark	no	yes	Explanatory text to further clarify the meaning of the usage of the product characterization class
Its superclass	no	no	Reference to the possible superclass of the product characterization class in the is-a hierarchy
Applicable properties	no	no	References to the new properties that become applicable to the product characterization class, besides the inherited applicable properties
Applicable types	no	no	References to the possible new named types that become applicable to the product characterization class (see 5.3)
Figure	no	no	Illustration to clarify the meaning of the definition of the product characterization class
Subclass selectors	no	no	Reference to non-quantitative code type properties that may be used for selecting subclasses (see 5.7)
Class selector values	no	no	Assignment of values to class selectors allowing to select the current product characterization class (see 5.7)

[a] "Mand": is the attribute mandatory or not?

[b] "Trans": can the content of the attribute be translated or not?