INTERNATIONAL STANDARD

ISO 21127

First edition
2006-09-15

Information and documentation — A reference ontology for the interchange of cultural heritage information

Information et documentation — Une ontologie de référence pour l'échange d'informations du patrimoine culturel

Reference number
ISO 21127:2006(E)

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.

The main task of technical committees is to prepare International Standards. Draft International Standards adopted by the technical committees are circulated to the member bodies for voting. Publication as an International Standard 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 21127 was prepared by Technical Committee ISO/TC 46, *Information and documentation*, Subcommittee SC 4, *Technical interoperability*, in collaboration with the International Council of Museums Committee for Documentation (ICOM CIDOC).

Introduction

This International Standard is the culmination of more than a decade of standards development work by the International Committee for Documentation (CIDOC) of the International Council of Museums (ICOM). Work on this International Standard began in 1996 under the auspices of the ICOM-CIDOC Documentation Standards Working Group. Throughout its development, the model has been known as the “CIDOC Conceptual Reference Model” or CRM. References to the CRM can be considered throughout as synonymous with ISO 21127.

The primary purpose of this International Standard is to offer a conceptual basis for the mediation of information between cultural heritage organizations such as museums, libraries and archives. This International Standard aims to provide a common reference point against which divergent and incompatible sources of information can be compared and, ultimately, harmonized.

ISO 21127 is a domain ontology [1)] for cultural heritage information: a formal representation of the conceptual scheme, or “world view”, underlying the database applications and documentation systems that are used by cultural heritage institutions. It is important to note that this International Standard aims to clarify the logic of what cultural heritage institutions do in fact document; it is not intended as a normative specification of what they *should* document. The primary role of this International Standard is to enable information exchange and integration between heterogeneous sources of cultural heritage information. It aims to provide the semantic definitions and clarifications needed to transform disparate, localized information sources into a coherent global resource, be it within an institution, an intranet or on the Internet.

The specific aims of this International Standard are to:

— Serve as a common language for domain experts and IT developers when formulating requirements.

— Serve as a formal language for the identification of common information contents in different data formats; in particular to support the implementation of automatic data transformation algorithms from local to global data structures without loss of meaning. These transformation algorithms are useful for data exchange, data migration from legacy systems, data information integration, and mediation of heterogeneous sources.

— Support associative queries against integrated resources by providing a global model of the basic classes and their associations to formulate such queries.

— Provide developers of information systems with a guide to good practice in conceptual modelling.

The CRM ontology is expressed as a series of interrelated concepts with definitions. This presentation is similar to that used for a thesaurus. However, the ontology is not intended as a terminology standard and does not set out to define the terms that are typically used as data in cultural heritage documentation. Although the presentation provided here is complete, it is an intentionally compact and concise presentation of the ontology's 80 classes and 130 unique properties. It does not attempt to articulate the inheritance of properties by subclasses throughout the class hierarchy (this would require the declaration of several thousand properties, as opposed to 130). However, this definition does contain all the information needed to infer and automatically generate a full declaration of all properties, including inherited properties.

1) In the sense used in computer science, i.e. it describes in a formal language the relevant explicit and implicit concepts and the relationships between them [1].

Information and documentation — A reference ontology for the interchange of cultural heritage information

1 Scope

This International Standard establishes guidelines for the exchange of information between cultural heritage institutions. In simple terms this can be defined as the curated knowledge of museums.

A more detailed definition can be articulated by defining both the intended scope, a broad and maximally inclusive definition of general principles, and the practical scope, which is defined by reference to a set of specific museum documentation standards and practices.

The intended scope of this International Standard is defined as the exchange and integration of heterogeneous scientific documentation relating to museum collections. This definition requires further elaboration:

— The term “scientific documentation” is intended to convey the requirement that the depth and quality of descriptive information that can be handled by this International Standard need be sufficient for serious academic research. This does not mean that information intended for presentation to members of the general public is excluded, but rather that this International Standard is intended to provide the level of detail and precision expected and required by museum professionals and researchers in the field.

— The term “museum collections” is intended to cover all types of material collected and displayed by museums and related institutions, as defined by ICOM[2]. This includes collections, sites, and monuments relating to fields such as social history, ethnography, archaeology, fine and applied arts, natural history, history of sciences and technology.

— The documentation of collections includes the detailed description of individual items within collections, groups of items and collections as a whole. This International Standard is specifically intended to cover contextual information (i.e. the historical, geographical, and theoretical background that gives museum collections much of their cultural significance and value).

— The exchange of relevant information with libraries and archives, and harmonization with their models, falls within the intended scope of this International Standard.

— Information required solely for the administration and management of cultural institutions, such as information relating to personnel, accounting and visitor statistics, falls outside the intended scope of this International Standard.

The practical scope[3] of this International Standard is the set of reference standards for museum documentation that have been used to guide and validate its development. This International Standard covers the same domain of discourse as the union of these reference documents; this means that data correctly encoded according to any of these reference documents can be expressed in a compatible form, without any loss of meaning.

2) The ICOM Statutes provide a definition of the term “museum” at <http://icom.museum/statutes.html#2>.

3) The practical scope of the CIDOC CRM, including a list of the relevant museum documentation standards, is discussed in more detail on the CIDOC CRM website at <http://cidoc.ics.forth.gr/scope.html>.

2 Conformance

Users intending to take advantage of the semantic interoperability offered by this International Standard should ensure conformance with the relevant data structures. Conformance pertains either to data to be made accessible in an integrated environment, or to contents intended for transport to other environments. Any encoding of data in a formal language that preserves the relations of the classes, properties and inheritance rules defined by this International Standard is regarded as conformant.

Conformance with this International Standard does not require complete matching of all local documentation structures, nor that all concepts and structures present in this International Standard be implemented. This International Standard is intended to allow room both for extensions, needed to capture the full richness of cultural information, and for simplification, in the interests of economy. A system will be deemed partially conformant if it supports a subset of subclasses and subproperties defined by this International Standard. Designers of the system should publish details of the constructs that are supported.

The focus of this International Standard is on the transport and mediation of structured information. It does not provide or require interpretation of unstructured free-text information into a structured, logical form. Free-text information, while supported, falls outside the scope of conformance considerations.

Any documentation system will be deemed conformant with this International Standard, regardless of the internal data structures it uses, if a deterministic logical algorithm can be constructed that transforms data contained in the system into a directly compatible form without loss of meaning. No assumptions are made as to the nature of this algorithm. “Without loss of meaning” signifies that designers and users of the system are satisfied that the data representation corresponds to the semantic definitions provided by this International Standard.

3 Terms and definitions

For the purposes of this document, the following terms and definitions apply. We have selected these terms for ease of understanding by non-computer experts from the various terminologies in use for object-oriented models.

3.1
class
category of items that share one or more common properties

NOTE Class properties serve as criteria to identify items that belong to the class. These properties need not be explicitly formulated in logical terms, but can be described in a text (called a scope note) that refers to a common conceptualisation of domain experts. The sum of these properties is called the intension of the class. A class can be the domain or range of none, one, or more properties formally defined in a model. The formally defined properties need not be part of the intension of their domains or ranges: such properties are optional. An item that belongs to a class is called an instance of this class. A class is associated with an open set of real-life instances, known as the extension of the class. Here “open” is used in the sense that it is generally beyond our capabilities to know all instances of a class in the world and, indeed, that the future can bring new instances into being at any time (Open World). Therefore a class cannot be defined by enumerating its instances. A class plays a role analogous to a grammatical *noun*, and can be completely defined without reference to any other construct (unlike properties, which need to have an unambiguously defined domain and range). For example, “Person” is a class. A “Person” can have the property of being a *member of* a “Group”, but this is not a necessary condition for being a “Person”. We will never know all “Persons” who have lived in the past, and there will be more “Persons” in the future. Classes are usually organized as a *class hierarchy*. The relationship between a subclass and its superclass is known as the *IsA* relationship (a concatenation of the words “is a”). For example, a ship *IsA* vehicle.

3.2
complement
⟨of a class A⟩ set of all instances of its superclass, B, that are not instances of class A

NOTE In terms of set theory, the complement of a class is the extension of the superclass minus the extension of the class. Compatible extensions of this International Standard need not declare any class as the complement of one or more other classes. To do so would violate the goal of describing an Open World. For example, for all possible cases of human *gender,* “male” need not be declared as the complement of “female” or vice versa.

3.3
disjoint
having no common instances in any possible world

NOTE 1 Classes are disjoint if the intersection of their extensions is necessarily an empty set.

NOTE 2 See also 5.4.

3.4
domain
class for which a property is formally defined

NOTE Instances of a property are applicable to instances of its domain class. A property needs to have exactly one domain, though the domain class can always contain instances for which the property is not instantiated. The domain class is analogous to the grammatical *subject* of a phrase while the property is analogous to the *verb*. Which class is selected as the domain and which as the range is arbitrary, as is the choice between active or passive voice. Property names in the CRM are designed to be semantically meaningful and grammatically correct when read from domain to range. The inverse property name, given in parentheses, is also designed to be semantically meaningful and grammatically correct when read from range to domain.

3.5
extension
set of all real life instances belonging to a class that fulfil the criteria of its intension

NOTE 1 The extension of a class is an "open" set in the sense that it is generally beyond our capabilities to know all instances of a class in the world. The future can bring new instances into being at any time (Open World). An information system may at any point in time refer to some instances of a class, which form a subset of its extension.

NOTE 2 See also 5.6.

3.6
inheritance
duplication of properties from a class to its subclasses

NOTE Inheritance of properties from superclasses to subclasses entails that if an item x is an instance of a class A, then all properties that need hold for the instances of any of the superclasses of A need also hold for item x, and that all optional properties that can hold for the instances of any of the superclasses of A can also hold for item x.

3.7
instance
item having properties that meet the criteria of the intension of the class

NOTE "The Mona Lisa" is an instance of the class of "physical man-made objects". An instance of a property is a factual relation between an instance of the domain and an instance of the range of the property that matches the criteria of the intension of the property. For example, "the Louvre *is current owner of* the Mona Lisa" is an instance of the property "is current owner of". The number of instances of a class declared in an information system is usually less than the total number of instances in the real world. For example, although *you* are an instance of "person", you are not mentioned in all information systems describing "persons".

3.8
intension
intended meaning of a class

NOTE The intension of a class consists of one or more common properties, or traits shared by all instances of the class. These properties need not be explicitly formulated in logical terms, but can simply be described in a text (a scope note) that refers to a conceptualization shared by domain experts.

3.9
interoperability
capability of different information systems to communicate some of their contents

NOTE Interoperability can imply that

a) two systems can exchange information, and/or

b) multiple systems can be accessed with a single method.

Generally, *syntactic* interoperability is distinguished from *semantic* interoperability. *Syntactic* interoperability means that the information encoding and the access protocols of the relevant systems are compatible, so that information can be processed as described above without error. However, syntactic interoperability alone does not ensure that each system processes the data in a manner consistent with the intended meaning. For example, one system can use a table called “Actor” and another one called “Agent”. Data from the two tables might remain separated, even though they can have exactly the same meaning. To overcome this situation, *semantic* interoperability has to be added. The CRM relies on existing *syntactic* interoperability and is concerned only with adding *semantic* interoperability.

3.10
monotonic
⟨of a knowledge base⟩ having a set of conclusions derived via inference rules that does not reduce, irrespective of the whatever additional propositions can be inserted

NOTE 1 Monotonic reasoning is a term derived from knowledge representation. In practical terms, as experts enter correct statements to an information system, the system need not regard any of the existing statements as invalid. The CRM ontology is designed for monotonic reasoning and so enables conflict-free merging of huge stores of knowledge.

NOTE 2 See also 5.1.

3.11
multiple inheritance
possibility for a class to have more than one immediate superclass

NOTE The extension of a class with multiple immediate superclasses is a subset of the intersection of all extensions of its superclasses. The intension of a class with multiple immediate superclasses extends the intensions of all its superclasses, i.e. its *traits* are more restrictive than any of its superclasses. If multiple inheritance is used, the resulting “class hierarchy” is a directed graph and not a tree structure. If it is represented as an indented list, then some classes will inevitably be repeated at different positions in the hierarchy. For example, “person” is both an “actor” and a “biological object”.

3.12
open world
assumption that the information stored in a knowledge base is incomplete with respect to the universe of discourse it aims to describe

NOTE A term derived from knowledge representation. The incompleteness of a knowledge base can be due to the inability of the maintainer to provide sufficient information, or to more fundamental problems of cognition in the system’s domain. Such problems are characteristic of cultural information systems since our records about the past are necessarily incomplete. In addition, some items cannot be clearly assigned to a given class. In particular, the absence of a certain property for an item described in the system does not necessarily entail that the item does not possess the property. For example, if one item is described as “biological object” and another as “physical object”, this does not imply that the latter is not also a “biological object”. Therefore, complements of a class with respect to a superclass cannot be *derived* in general from an information system based on the open world assumption.

3.13
primitive concept
concept that is declared and for which the meaning is clear, but which cannot be derived from other concepts

NOTE Primitive concept is a term derived from knowledge representation. For example, *mother* can be described as a female who has given birth to a child, so *mother* is not a primitive concept. *Event* however is a primitive concept. The CRM is composed primarily of primitive concepts.

3.14
property
defining characteristic that serves to define a relationship of a specific kind between two classes

NOTE A property is characterized by an intension, which is conveyed by a scope note. A property plays a role analogous to a *verb* in that it need be defined with reference to both a domain and range, which are analogous to the *subject* and *object* in a phrase (unlike classes, which can be defined independently). Which class is selected as the domain and which as the range, is arbitrary, as is the choice between active and passive voice. In other words, a property can be interpreted in both directions, with two distinct but related interpretations. Properties can themselves have properties that relate to other classes. (This feature is used in this model only in order to describe dynamic subtyping of properties.) Properties can also be *specialized* in the same manner as classes, resulting in *IsA* relationships between subproperties and their superproperties. For example, “physical man-made thing *depicts* CRM entity” is equivalent to “CRM entity *is depicted by* physical man-made thing”.

3.15
query containment
query X contains another query Y if, for each possible population of a database, the answer set to query X also contains the answer set to query Y

NOTE If query X and Y were classes, then X would be a superclass of Y.

3.16
range
class that comprises all the potential values of a property

NOTE Instances of a property can only link to instances of its range class. A property needs to have exactly one range, though the range class can also contain instances that are not values of the property. The range class is analogous to the grammatical *object* of a phrase, while the property is analogous to the *verb*. Which class is selected as domain, and which as range, is arbitrary, as is the choice between active and passive voice. Property names in the CRM are designed to be semantically meaningful and grammatically correct when read from domain to range. The inverse property name, given in parentheses, is designed to be semantically meaningful and grammatically correct when read from range to domain.

3.17
scope note
textual description of the intension of a class or property

NOTE Scope notes are not formal modelling constructs but are provided to help explain the intended meaning and application of the CRM’s classes and properties. Basically, they refer to a conceptualization shared by domain experts and disambiguate different possible interpretations. Illustrative examples of classes and properties are also provided with the scope notes for explanatory purposes.

3.18
shortcut
formally defined single property that represents a *deduction* or *join* of a data path in the CRM

NOTE 1 The scope notes of shortcut properties provide a verbal description of the equivalent deduction. Shortcuts are introduced for those cases where common documentation practice refers only to the deduction rather than to the fully developed path. For example, museums often only record the “dimension” of an object without documenting the E16 measurement activity that observed it. The CRM allows shortcuts as cases of less detailed knowledge, while preserving in its schema the relationship to the full information.

NOTE 2 See also 5.3.

3.19
strict inheritance
properties inheritance that allows no exceptions

NOTE Some systems can declare that “elephants are grey”, and regard a white elephant as an exception. Under strict inheritance rules it would hold that if all elephants were indeed grey, then a white elephant could not be an elephant. Obviously not all elephants are grey; being grey is not part of the intension of the concept elephant but an optional property. The CRM applies strict inheritance as a normalization principle.

3.20
subclass
specialization of another class, i.e. the superclass

NOTE A subclass inherits *all* the properties of its superclass (i.e. strict inheritance), in addition to having none, one, or more additional properties of its own. A subclass can have more than one immediate superclass, and consequently inherits the properties of all of its superclasses (i.e. multiple inheritance). A subclass has an *IsA* relationship to its superclass(es): every instance of the subclass is also, by definition, an instance of the superclass(es). For example, every "person" *IsA* "biological object".

3.21
subproperty
specialization of another property, i.e. the superproperty

NOTE 1 All instances of a subproperty are also instances of its superproperty. The intension of a subproperty extends the intension of its superproperty, i.e. its *traits* are more restrictive than that of its superproperty. The domain of a subproperty is a subclass of the domain of its superproperty. The range of a subproperty is a subclass of the range of its superproperty. Instances of a subproperty inherit the definition of *all* of the properties declared for its superproperty without exceptions (strict inheritance), in addition to having none, one, or more properties of their own.

NOTE 2 A subproperty can have more than one immediate superproperty and consequently inherits the properties of all of its superproperties (multiple inheritance). The *IsA relationship* or *specialization* between two or more properties gives rise to the structure we call a *property hierarchy.* The *IsA relationship* is transitive and can not be cyclic. In some object-oriented languages, including C++, there is no equivalent to the specialization of properties.

3.22
superclass
generalization of one or more other classes, i.e. the subclasses

NOTE A superclass *subsumes* all instances of its subclasses, and can also have additional instances that do not belong to any of its subclasses. The intension of the superclass is less restrictive than any of its subclasses. The *subsumption relationship* or *generalization* is the inverse of the *IsA relationship* or *specialization*. In some contexts (e.g. the programming language C++) the term *parent class* is used synonymously with superclass. For example, "biological object *subsumes* person" is synonymous with "biological object is a *superclass* of person". Fewer properties are needed to identify an item as a "biological object" than to identify it as a "person".

3.23
superproperty
generalization of one or more other properties, i.e. the subproperties

NOTE A superproperty *subsumes* all instances of its subproperties, and can also have additional instances that do not belong to any of its subproperties. The intension of the superproperty is less restrictive than any of its subproperties. The *subsumption relationship* or *generalization* is the inverse of the *IsA relationship* or *specialization*.

4 Structure and presentation

4.1 Property quantifiers

Quantifiers for properties are provided for the purpose of semantic clarification only, and should *not* be treated as implementation recommendations. This International Standard has been designed to accommodate alternative opinions and incomplete information; *all* properties should therefore be implemented as optional and repeatable for their domain and range ("many to many $(0,n:0,n)$"). The term "cardinality constraints" is avoided here as it typically pertains to implementations.

Table 1 lists all possible property quantifiers occurring in this document according to their notation, together with a textual explanation. In order to provide optimal clarity, two widely accepted notations are used in this International Standard, i.e. one verbal, the other numerical. The verbal notation uses phrases such as "one to many", and the numerical notation expressions such as "$(0,n:0,1)$". The terms "one", "many" and "necessary" are fairly intuitive; the term "dependent" is less obvious. It denotes a situation where a range instance cannot exist without an instance of the respective property. In other words, the property is "necessary" for its range.

Table 1 — Property quantifiers

Quantifier	Description
many to many (0,*n*:0,*n*)	Unconstrained: an individual domain instance and range instance of this property can have zero, one, or more instances of the property. In other words, the property is optional and repeatable for its domain and range.
one to many (0,*n*:0,1)	An individual domain instance of this property can have zero, one, or more instances of the property, but an individual range instance cannot be referenced by more than one instance of this property. In other words, the property is optional for its domain and range, but repeatable for its domain only. This situation is sometimes called a “fan-out”.
many to one (0,1:0,*n*)	An individual domain instance of this property can have zero or one instance of the property, but an individual range instance can be referenced by zero, one, or more instances of the property. In other words, the property is optional for its domain and range, but repeatable for its range only. This situation is sometimes called a “fan-in”.
many to many, necessary (1,*n*:0,*n*)	An individual domain instance of this property can have one or more instances of the property, but an individual range instance can have zero, one, or more instances of the property. In other words, the property is necessary and repeatable for its domain, and optional and repeatable for its range.
one to many, necessary (1,*n*:0,1)	An individual domain instance of this property can have one or more instances of the property, but an individual range instance cannot be referenced by more than one instance of the property. In other words, the property is necessary and repeatable for its domain, and optional but not repeatable for its range. This situation is sometimes called a “fan-out”.
many to one, necessary (1,1:0,*n*)	An individual domain instance of this property shall have exactly one instance of the property, but an individual range instance can be referenced by zero, one, or more instances of the property. In other words, the property is necessary and not repeatable for its domain, and optional and repeatable for its range. This situation is sometimes called a “fan-in”.
one to many, dependent (0,*n*:1,1)	An individual domain instance of this property can have zero, one, or more instances of the property, but an individual range instance shall be referenced by exactly one instance of the property. In other words, this property is optional and repeatable for its domain, but necessary and not repeatable for its range. This situation is sometimes called a “fan-out”.
one to many, necessary, dependent (1,*n*:1,1)	An individual domain instance of this property can have one or more instances of the property, but an individual range instance shall be referenced by exactly one instance of the property. In other words, the property is necessary and repeatable for its domain, and necessary but not repeatable for its range. This situation is sometimes called a “fan-out”
many to one, necessary, dependent (1,1:1,*n*)	An individual domain instance of this property shall have exactly one instance of the property, but an individual range instance can be referenced by one or more instances of the property. In other words, this property is necessary and not repeatable for its domain, and necessary and repeatable for its range. This situation is sometimes called a “fan-in”
one to one (1,1:1,1)	An individual domain instance and range instance of this property shall have exactly one instance of the property. In other words, the property is necessary and not repeatable for its domain and for its range.

NOTE Some properties are defined as being necessary for their domain or as being dependent on their range. If such properties are not specified for an instance of the respective domain or range, it means that the property exists, but that the value on one side of the property is unknown. In the case of optional properties, no distinction is made between a value being unknown or the property not being applicable at all. For example, one can know that an object has an owner, but not know who the owner is, or know that an object has no owner. The model makes no distinction between these two cases. A textual note can be used for clarification if needed.

4.2 Naming conventions

The following naming conventions have been applied hereafter:

— Classes are identified by numbers [4] preceded by the letter "E" (historically classes were sometimes referred to as "Entities"), and are named using noun phrases (nominal groups) in title case (initial capitals). For example, *E63 Beginning of Existence*.

— Properties are identified by numbers preceded by the letter "P," and are named in both directions, using verbal phrases in lower case. Properties with the character of states are named in the present tense, such as "has type", whereas properties relating to events are named in past tense, such as "carried out". For example, *P126 employed (was employed by)*.

— Property names should be read in their non-parenthetical form for the domain-to-range direction, and in parenthetical form for the range-to-domain direction.

— Properties with a range that is a subclass of *E59 Primitive Value* (such as *E1 CRM Entity.P2 has note: E62 String*) have no parenthetical name form as reading the property name in the range-to-domain direction is not regarded as meaningful.

— Properties that have identical domain and range are either symmetric or transitive. Instantiating a symmetric property implies that the relation holds for both the domain-to-range and the range-to-domain directions. An example of this is *E53 Place.P122 borders with: E53 Place*. The names of symmetric properties have no parenthetical form, because reading in the range-to-domain direction is the same as the domain-to-range reading. Transitive asymmetric properties, such as *E4 Period. P9 consists of (forms part of): E4 Period*, do have a parenthetical form that relates to the meaning of the inverse direction.

— The choice of property domains, and hence the order of their names, is established in accordance with the following priority list:

1) Temporal Entity and its subclasses;

2) Thing and its subclasses;

3) Actor and its subclasses;

4) Other.

5 Modelling principles

5.1 Monotonicity

Because this International Standard's primary role is the meaningful integration of information in an Open World*,* it aims to be monotonic in the sense of Domain Theory. Existing constructs, and deductions made from them, shall always remain valid and well-formed, i.e. even if new constructs and extensions are added.

For example, one may add a subclass of *E7 Activity* to describe the use of a certain name for a place over a certain time-span by a particular group. By this extension, no existing *IsA Relationships* or property inheritances are compromised.

In addition, this International Standard aims to enable the formal preservation of monotonicity when augmenting a compatible system. Existing instances, their properties, and deductions made from them, should always remain valid and well-formed even as new instances are added to the system.

---

4) Some gaps are present in the numbering sequence used for classes are properties. This is intentional: numbers assigned in previous versions of the CRM to deprecated classes and properties have not been re-used.

For example, if someone describes correctly that an item is an instance of *E19 Physical Object* and, subsequently, it is correctly characterized as an instance of *E20 Biological Object*, the system should not stop treating it as an instance of *E19 Physical Object*.

In order to formally preserve monotonicity in cases where opinions diverge, all formally defined properties should be implemented with unconstrained cardinality (many:many) so that conflicting instances of properties are merely accumulated. Knowledge stored in a conformant system can thus serve as a research base, accumulating relevant alternative opinions around well-defined entities. Conclusions about the truth or falsehood of the instances stored remain the subject of open-ended scientific or scholarly hypothesis building.

For example, “El Greco” and even “King Arthur” should be treated as instances of *E21 Person* and be dealt with as existing within the domain of discourse once they are entered into a knowledge base. Alternative opinions about properties, such as their birthplace and the details of their lives, may be accumulated without decisions concerning their veracity being required during data compilation.

5.2 Minimality

Although the scope of this International Standard is very broad, the ontology itself is constructed as economically as possible.

— A class is not declared unless it is required as the domain or range of a property not appropriate to its superclass, or it is a key concept in the *practical scope*.

— Classes and properties that share a superclass are *non-exclusive* by default. For example, an object may be both an instance of *E20 Biological Object* and *E22 Man-made Object*.

— Classes and properties are either primitive, or constitute key concepts in the practical scope.

— Complements of classes are not declared.

5.3 Shortcuts

Some properties are declared as shortcuts of longer, more comprehensively articulated paths that connect the same domain and range classes as the shortcut property via one or more intermediate classes. For example, the property *E18 Physical Thing. P52 has current owner: E39 Actor*, is a shortcut for a fully articulated path from *E18 Physical Thing* through *E8 Acquisition* to *E39 Actor*. An instance of the fully-articulated path always implies an instance of the shortcut property. However, the inverse may not be true; an instance of the fully-articulated path cannot always be inferred from an instance of the shortcut property.

5.4 Disjointness

Classes are disjoint if they share no common instances in any possible world. There are many examples of disjoint classes in the CRM.

A comprehensive declaration of all possible disjoint class combinations afforded by the CRM has not been provided here; it would be of questionable practical utility and would easily become inconsistent with the goal of providing a concise definition. However, the two following examples of disjoint class pairs are fundamental to an effective comprehension of the CRM ontology.

5.4.1 E2 Temporal Entity is disjoint from E77 Persistent Item

Instances of the class *E2 Temporal Entity* perdure, whereas instances of the class *E77 Persistent Item* endure. Even though instances of *E77 Persistent Item* have a limited existence in time, they are fundamentally different in nature from instances of *E2 Temporal Entity* because they preserve their identity between events. Declaring enduring and perduring entities as disjoint classes is consistent with the distinctions made in data structures that fall within the CRM’s practical scope.