ID:

571

Starting Date:

2021-11-12

Working Group:

3

Status:

Open

Background:

In the 51st CIDOC CRM & 44th FRBRoo SIG meeting, upon discussing issue [532](http://www.cidoc-crm.org/Issue/ID-532-the-cardinality-of-ap13.2-is-justified-by-is-justification-of), the SIG decided to start a new issue, where to discuss the section that cardinalities of typed properties (.1/.2) should appear in the CIDOC CRM document. If cardinalities for typed properties are not to be defined on a one-by-one basis, this calls for drafting a text settling them in the introduction section. Otherwise, a template is needed for adding cardinality constraints to the definitions of typed properties, for the properties that have them.

## Reference to Issues:

532

[The cardinality of AP13.2 is justified by (is justification of)](https://cidoc-crm.org/Issue/ID-532-the-cardinality-of-ap13.2-is-justified-by-is-justification-of)

**Elaboration:**

Under “***About the logical expressions used in the CIDOC CRM***”, we find these paragraphs:

The present CIDOC CRM specifications are annotated with logical axioms, providing an additional formal expression of the CIDOC CRM ontology. This section briefly introduces the assumptions that are at the basis of the logical expression of the CIDOC CRM (for a fully detailed account of the logical expression of semantic data modelling, see (Reiter,1984)).

The CIDOC CRM is expressed in terms of the primitives of semantic data modelling. As such, it consists of:

· *classes,* which represent general notions in the domain of discourse, such as the CIDOC CRM class E21 Person which represents the notion of person;

· *properties,* which represent the binary relations that link the individuals in the domain of discourse, such as the CIDOC CRM property *P152 has parent* linking a person to one of the person’s parent.

Classes and properties are used to express ontological knowledge by means of various kinds of constraints, such as sub-class/sub-property links, e.g., E21 Personis a sub-class ofE20 Biological Object, or domain/range constraints, e.g., the domain of *P152 has parent* is class E21 Person*.*

In contrast, first-order logic-based knowledge representation relies on a language for formally encoding an ontology. This language can be directly put in correspondence with semantic data modelling in a straightforward way:

· classes are named by *unary predicate symbols*; conventionally, we use E21 as the unary predicate symbol corresponding to class E21 Person;

· properties are named by *binary predicate symbols*; conventionally, we use P152 as the binary predicate symbol corresponding to property *P152 has parent.*

· properties of properties, “.1 properties” are named by *ternary predicate symbols*; conventionally, we use P14.1 as the ternary predicate symbol corresponding to property *P14.1 in the role of.*

—---------------------------------------------------------

Under “About Types”, we find this paragraph:

Analogous to the function of the *P2 has type (is type of)* property, some properties in the CIDOC CRM are associated with an additional property. These are numbered in the CIDOC CRM documentation with a ‘.1’ extension. The range of these properties of properties always falls under E55 Type. The purpose of a property of a property is to provide an alternative mechanism to specialize its domain property through the use of property subtypes declared as instances of E55 Type. They do not appear in the property hierarchy list but are included as part of the property declarations and referred to in the class declarations. For example, *P62.1 mode of depiction*: E55 Type is associated with E24 Physical Man-made Thing. *P62 depicts (is depicted by):* E1 CRM Entity.

We **completely miss** a statement about the actual form of the CRM **under “Applied Form”.**

Instead, it is implicit in “***About the logical expressions used in the CIDOC CRM***”, as shown above, and misses properties of properties.

**I propose:**

## **Applied Form**

The CIDOC CRM is an ontology in the sense used in computer science. It has been expressed as an object-oriented semantic model, in the hope that this formulation will be comprehensible to both documentation experts and information scientists alike, while at the same time being readily converted to machine-readable formats such as RDF Schema or OWL. A CRM conformant documentation system can be implemented using RDF Schema or OWL, but also in Relational or Object-Oriented schema. CIDOC CRM instances can be encoded in RDF, JSON LD, XML, OWL and others.

More specifically, the CIDOC CRM is expressed in terms of the primitives of semantic data modelling. As such, it consists of:

· *classes,* which represent general notions in the domain of discourse, such as the CIDOC CRM class E21 Person which represents the notion of person;

· *properties,* which represent the binary relations that link the individuals in the domain of discourse, such as the CIDOC CRM property *P152 has parent* linking a person to one of the person’s parent.

* properties of properties, such as the property *P14.1 in the role of*  of the CIDOC CRMproperty *P14 carried out by* (see also section “About Types”)*.* (They do not appear in the property hierarchy list, but are included as part of their base property declaration and are referred to in the class declarations. They all have the implicit quantification “many to many” (see also section “Property Quantifiers”))

Although the definition of the CIDOC CRM provided here is complete, it is an intentionally compact and concise presentation of the CIDOC CRM’s 81 classes and 160 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 160). However, this definition does contain all of the information necessary to infer and automatically generate a full declaration of all properties, including inherited properties.

—-------------------------------------------------------------------------------------

Add to ***Naming Conventions:*** a last bullet item:

· Properties of properties are identified by “P”, followed by the number of the base property extended with “.1” and are named in one direction using a verbal phrase in lower case in the present tense. For example: the property *P62.1 mode of depiction* of the property *P62 depicts (is depicted by).*

—--------------------------------------------------------------------------------------

***About the logical expressions used in the CIDOC CRM***”

The present CIDOC CRM specifications are annotated with logical axioms, providing an additional formal expression of the CIDOC CRM ontology. This section briefly introduces the assumptions that are at the basis of the logical expression of the CIDOC CRM (for a fully detailed account of the logical expression of semantic data modelling, see (Reiter,1984)).

In terms of semantic data modelling, classes and properties are used to express ontological knowledge by means of various kinds of constraints, such as sub-class/sub-property links, e.g., E21 Personis a sub-class ofE20 Biological Object, or domain/range constraints, e.g., the domain of *P152 has parent* is class E21 Person*.*

In contrast, first-order logic-based knowledge representation relies on a language for formally encoding an ontology. This language can be directly put in correspondence with semantic data modelling in a straightforward way:

· classes are named by *unary predicate symbols*; conventionally, we use E21 as the unary predicate symbol corresponding to class E21 Person;

· properties are named by *binary predicate symbols*; conventionally, we use P152 as the binary predicate symbol corresponding to property *P152 has parent.*

· properties of properties, “.1 properties”, are named by *ternary predicate symbols*; conventionally, we use P14.1 as the ternary predicate symbol corresponding to property *P14.1 in the role of.*

………

—----------------------------------------------------------------------------------------------------------------

Add at the end of “***Property Quantifiers” :***

Note that the quantification of all properties of properties, “.1 properties”, is “many to many”, and therefore does not appear explicitly in their definitions.

—--------------------------------------------------------------------------------------------------------------------------