Modeling and Querying Provenance using CIDOC CRM

Maria Theodoridou, Yannis Tzitzikas, Martin Doerr, Yannis Marketakis, Valantis Melessanakis

Institute of Computer Science, FORTH-ICS
Crete, Greece
Emails: {maria | tzitzik | martin | marketak | melesan}@ics.forth.gr

Version: Draft 0.94
Date: December 18, 2008

Abstract
This paper elaborates on the problem of provenance for both physical and digital objects. In particular it discusses provenance according to OAIS (ISO 14721:2003) and how it relates with the conceptualization of CIDOC CRM ontology (ISO 21127:2006). Subsequently it introduces an extension of the CIDOC CRM able to capture the modeling and the query requirements regarding the provenance of digital objects. Over this extension the paper provides a number of indicative examples of modeling for provenance in various domains. Finally, a number of indicative provenance query templates are described.

1. Introduction

Provenance is the origin or source from which something comes, and the history of subsequent owners (also known in some fields as *chain of custody*). The term is often used in the sense of place and time of manufacture, production or discovery. Comparative techniques, expert opinion, written and verbal records and the results of tests are often used to help establish provenance.
The provenance of works of fine art, antiques and antiquities often assumes great importance. Documented evidence of provenance for an object can help to establish that it has not been altered and is not a forgery or a reproduction. Knowledge of provenance can help to assign the work to a known artist and a documented history can be of use in helping to prove ownership. The quality of provenance of an important work of art can make a considerable difference to its selling price in the market; this is affected by the degree of certainty of the provenance, the status of past owners as collectors, and in many cases by the strength of evidence that an object has not been illegally excavated or exported from another country.

The provenance of a work of art may be recorded in various forms depending on context or the amount that is known, from a single name to an entry in a full scholarly catalogue several thousand words long. Scientific research is generally held to be of good provenance when it is documented in detail sufficient to allow reproducibility.

Due to all these reason the provenance of digital objects has to be archived and this is also suggested by the OAIS standard. To this end, we need conceptual models able to capture the various forms of provenance information that we may have. Such models can enable the exchange and integration of provenance data and can guide the design of provenance services. The contribution of this paper lies in:

(a) describing an extension of CIDOC CRM for digital objects,
(b) providing examples of how provenance information can be modeled
(c) identifying a number of basic queries that could be used for reasoning about the provenance of digital objects.

The remainder of this paper is organized as follows: Section 2 discusses provenance from the perspective of both OAIS and CIDOC CRM while Section 3 describes the extension of CIDOC CRM for digital objects. Section 4 describes indicative provenance queries assuming CIDOC CRM. Section 5 provides specific modeling example coming from the area of Contemporary Performing Arts. Finally, Section 7 summarizes and concludes the paper.

2. Provenance and OAIS

In the context of an OAIS, provenance describes events that occur during a digital object's lifecycle. The Provenance Information documents the history of the Content Information. This information tells the origin or source of the Content Information, any changes that may have taken place since it was originated, and who has had custody of it since it was originated. Examples of Provenance Information are the principal investigator who recorded the data, and the information concerning its storage, handling, and migration.

The middle column of Table 1 shows examples OAIS Provenance (as listed in the standard) for various types of content information. The right column comments each row with respect to CIDOC CRM, while a more detailed discussion is given in the subsequent section.

Content Information Type	OAIS Provenance	CIDOC CRM Provenance
Space Science Data	• Instrument description • Processing history • Sensor description • Instrument • Instrument mode • Decommutation map • Software interface specification	• Context of observation/experiment • By whom, Derivation chain • Context of observation/experiment • Context of observation/experiment • Context of observation/experiment • Context of observation/experiment • Context of observation/experiment
Digital Library Collections	For scanned collections: • metadata about the digitisation process • pointer to master version For born-digital publications: • pointer to the digital original • Metadata about the preservation process: • pointers to earlier versions of the collection item • change history	For scanned collections: • Context of digitisation process • Derivation chain For born-digital publications: • Derivation chain • Context of preservation process: • Derivation chain • Derivation chain
Software Package	• Revision history • License holder • Registration • Copyright	• Derivation chain • By whom • By whom • By whom

Table 1 OAIS and CIDOC CRM Provenance

3. CIDOC CRM Extension for Digital Objects

In the context of CIDOC CRM provenance-related services could be considered as queries that can take as input an object and answer questions of the form:

- Context
  - by whom (either creator or responsible for creation)
  - of observation/experiment
  - of digitization
  - Relation to reality
- Derivation chain

The current version of CIDOC CRM (version 4.2) can support queries regarding the creator or the responsible for creation of an object ("*by whom*" type of queries). Examples are provided later on.

The other two types of provenance queries ("*context*" and "*derivation chain*" queries) are not directly supported by the current version. For this purpose below we describe some extensions for capturing such cases.

The extension of CIDOC CRM for digital objects, hereafter CIDOC CRM Digital, expressed in RDFS is available at: http://cidoc.ics.forth.gr/rdfs/caspar/cidoc_digital.rdfs.

Figure 1 depicts an overview of the extensions as visualized by StarLion [StarLion]. The diagram shows the new classes and new properties (in green) and the directly referred objects from CIDOC CRM (in yellow). Its representation in RDF is also available at Appendix A.

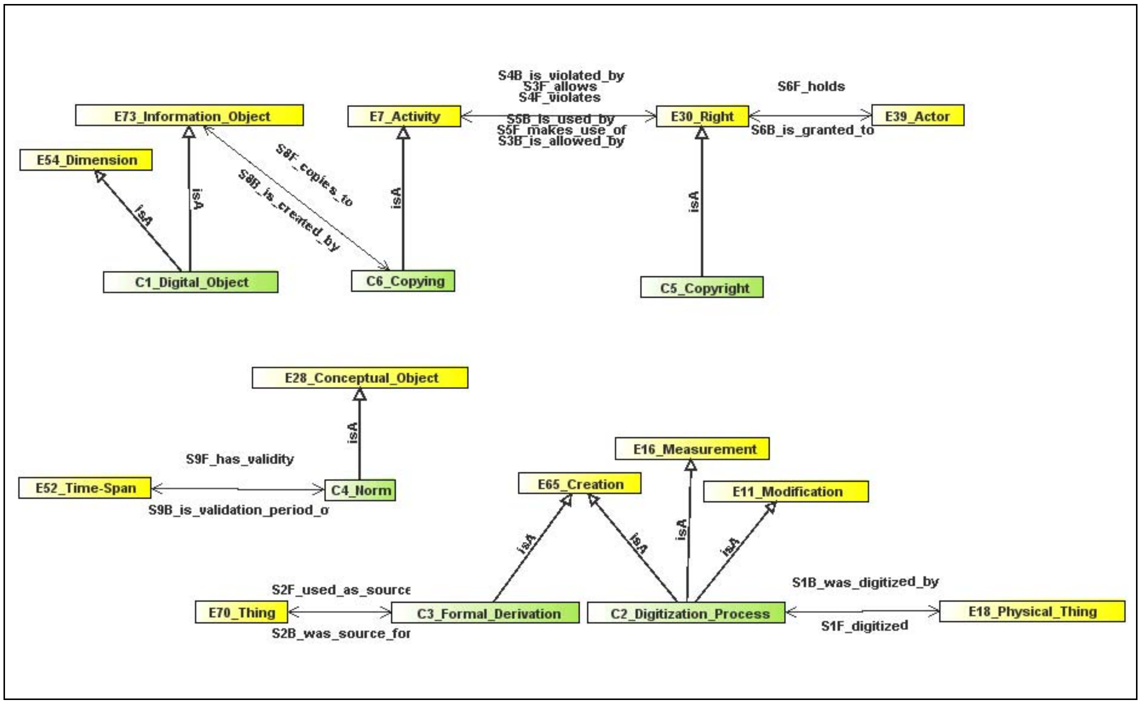


Figure 1 CIDOC CRM Digital

Specifically, the main classes that we defined in order to model **context** and **derivation chain** information are:

- ***C1 Digital Object***, which comprises identifiable immaterial items, that are composed of a set of bit sequences, such as data sets, e-texts, images, audio or video items, software, etc., that have an objectively recognizable structure and are documented as single units. A **C1 Digital Object** can represent things at any arbitrary level of granularity. This means that, for instance, it can represent a journal, an individual issue of a journal, an individual article in the journal, or a single table in that article. A **C1 Digital Object** does not depend on a specific physical carrier, and it can exist on one or more carriers simultaneously.

- ***C2 Digitization Process***, which comprises events that result in the creation of instances of **C1 Digital Object** that represent the appearance and/or form of an instance of **E18 Physical Thing** such as paper documents, archaeological objects, films, photographs, etc. It represents the transition from a material thing to an immaterial representation of it. The characteristic subsequent processing steps on digital objects are regarded as instances of **C3 Formal Derivation**.

- ***C3 Formal Derivation***, which comprises events that result in the creation of a **C1 Digital Object** from another one following a deterministic algorithm. The resulting instance of **C1 Digital Object** shares representative properties with the original object and can be mechanically reproduced. This class represents the transition from an immaterial object to another immaterial object.

Figure 2 presents a partial view of CIDOC CRM and the integration of the proposed extensions of classes **C2 Digitization Process** and **C1 Digital Object**. The heavy blue arrows represent the subclass hierarchy while black arrows show the relationships between the classes with the property names printed on the arrows. As can be seen, an **C2 Digitization Process** is defined as a subclass of activities **E65 Creation**, **E11 Modification** and **E16 Measurement** and is related to class **E70 Thing** through property ***S1 digited*** which is a subclass of ***P16 used specific object***. The relationships between **C1 Digital Object** and **C2 Digitization Process** are illustrated in Figure 3. An instance of **C2 Digitization Process** represents the transition from an instance of a material thing (**E18 Physical Thing**) to an instance of an immaterial representation of it (**C1 Digital Object**).

The value of **E55 Type** of property ***P16.1 mode of use*** of ***S1 digitized*** can be used to describe the source object of the Digitization Process and/or tools, instruments used. In this way, all the information regarding the digitization context and the derivation chain is present. In section 4 we present specific examples that use the described classes.

Digitization process: From a material to an immaterial object

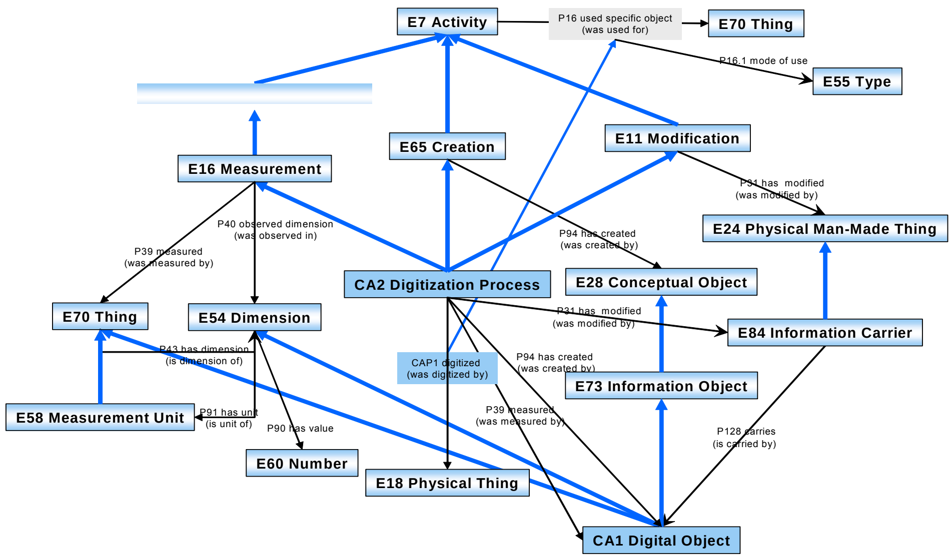


Figure 2: Digitization Process

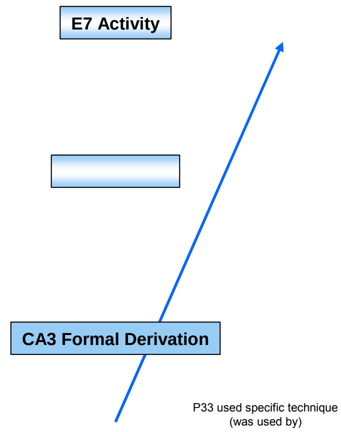


Figure 4 presents a partial view of CIDOC CRM and the integration of the proposed class **C3 Formal Derivation** which is defined as a subclass of **E65 Creation**. The property ***S2 used as source*** which is defined as a subclass of ***P16 used specific object (was used for)*** is a pointer to the original object. while properties ***P32 used general technique (was technique of)*** and ***P33 used specific technique (was used by)*** are used to specify the method, algorithm, software etc. used by the formal derivation. Property ***P3 has note*** between **C3 Formal Derivation** and **E62 String** is used to keep information regarding the property list used by the deterministic algorithm that the specific instance of **C3 Formal Derivation** used.

In Appendix A we present a detailed description of the classes described in this section following the CIDOC CRM class and property format.

4. Provenance Queries over CIDOC CRM

The part of the CIDOC CRM ontology that could be exploited for modeling the provenance of objects (either physical or digital) is shown next. Queries regarding provenance, could be based on paths of the above schema. Specifically we could identify the following query requirements

- Get the creator of an object
- Get the earlier versions of an item
- Get the events that changed the custody of an item
- Get the master version of an object
- Get the scanner/resolution of a digital object

The following table provides an indicative list of such queries. They could be considered as general purpose query templates that could be refined according to needs.

#	Description	Input	Output	Path
1	getCreatorOf	A Digital Object Instance of E28 Conceptual Object	Instances of E82 Actor Appellation	DO = E28 Conceptual Object→P94B was created by→E65 Creation→ P14F carried out by(→P14.1 in the role of→E55 Type=Creator)→ E39 Actor→P131F is identified by→E82 Actor Appellation

2	getScanner	A Digital Object Instance of E73 Information Object (Digital Image)	Instances of E70 Thing	DO=E73 Information Object→P39B was measured by→Digitization Process→ P16F used specific object→ (P16.1mode of use→E55 Type=Scanner)→E70 Thing
3	getResolution	A Digital Object Instance of E73 Information Object (Digital Image)	Instances of E60 Number	DO=E73 Information Object→P39B was measured by→ Digitization Process→P40F observed dimension→ E54 Dimension→P90F has value→E60 Number
4	getMasterVersion	A Digital Object Instance of E73 Information Object (Digital Image)	Instance of E70 Thing	DO=E73 Information Object→P94B was created by→ Digitization Process→digitized→E70 Thing
5	get earlier versions of the item	A Digital Derivative Instance of E29 Design or Procedure	Instance of E70 Thing	Item={E29 Design or Procedure→P94B was created by→E65 Creation→P15F was influenced by→E29 Design or Procedure}* repeat until P15F was influenced by is null
6	Events that change custody	A Digital Object Instance of E73 Information Object	List of Instances of E82 Actor Appellation	DO=E73 Information Object→P50F has current keeper→ {E39 Actor→P29B received custody through→E10 Transfer of Custody→ P28F custody surrendered by→E39Actor}* repeat until *P29B received custody through* is null → P131F is identified by→E82 Actor Appellation

Table 2 Provenance Query Templates over CIDOC CRM

Each template has a name, it takes as input a type-restricted resource (e.g. an instance of **E28 Conceptual Object**), and returns as output a number of typed resources (of course, as in any object oriented system, the type of the actual input/output parameters can be a subtype of the one specified in the template). For each template the path that should be followed for computing the answer is specified in the form of sequences of the form:

SourceClassName-PropertyName->TargetClassName

Some of these templates are recursive. For instance, consider template number 5:

Item={E29 Design or Procedure→P94B was created by→E65 Creation→P15F was influenced by→E29 Design or Procedure}*
repeat until P15F was influenced by is null

It takes as input a digital derivative, i.e. an instance of **E29**, and returns instances of **E70** (Thing). The query comprises an expression that takes as input an instance of **E29** and returns another instance(s) of **E29** (those influenced by) and this is continued recursively until the there is no other "**P15F was influenced by**" property that could be followed.

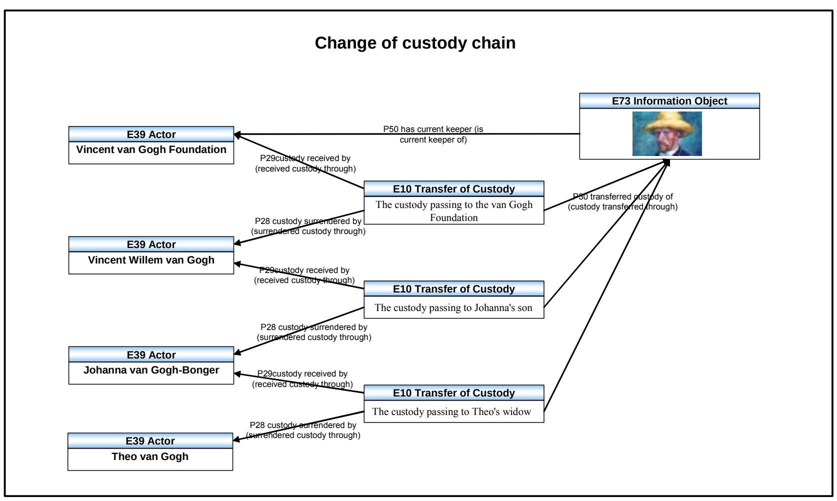


Figure 5 Sample query 1 – Find creator/producer

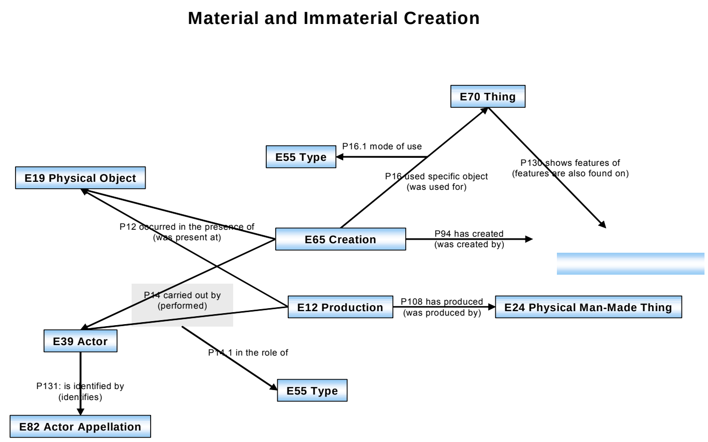


Figure 6: Sample query 6 – Change of custody chain

5. Provenance examples from various application domains

This section provides some indicative examples of modeling coming from various applications domains.

5.1. Example in the performing arts domain - Avis de Tempête

Figure 7 shows an example of modeling a performing art. Some provenance queries are described in the next table.

Who wrote the ADT libretto	Find E82 Actor Appellation *P131B identifies* (*P14.1 in the role of* E55 Type = "Writer") E39 Actor *P14B performed* E65 Creation *P94F was created by* E28Conceptual Object = "ADT"
Who wrote the ADT music	Find E82 Actor Appellation where *P131B identifies* (*P14.1 in the role of* E55 Type = "Composer") E39 Actor *P14B performed* E65 Creation *P94Fwas created by* E28Conceptual Object = "ADT"

5.2. *Example in the performing arts domain - Jupiter*

Figure 8 shows an example of modeling a performing art. Some provenance queries are described in the next table.

Which is the first version of the Jupiter's composition	Find E29 Design or Procedure where *P94B was created by* E65 Creation *P15F was influenced by* E29 Design or Procedure repeat until *P15F was influenced by* is null

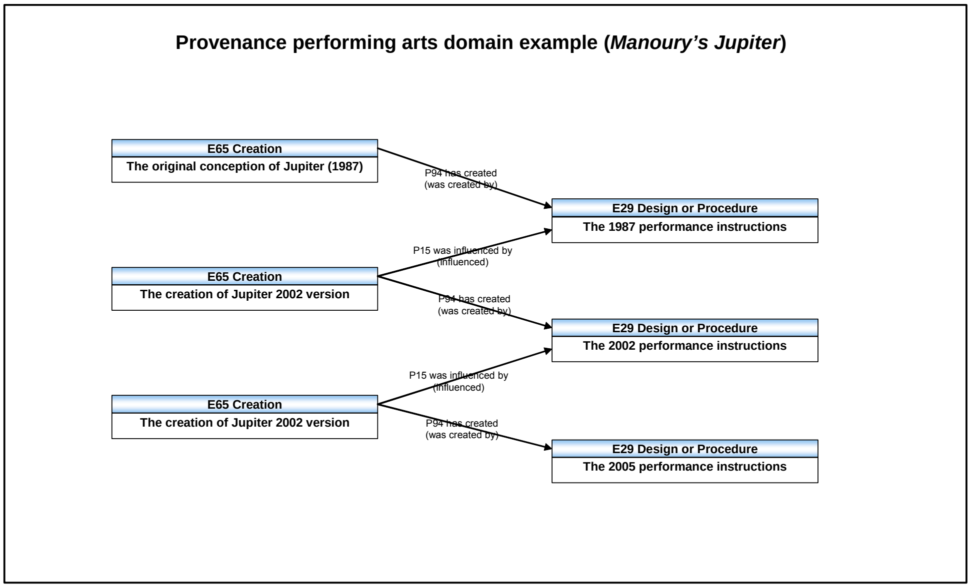


Figure 8: Jupiter

5.3. *Software Conversion Example*

Here we describe how we can model activities that result in the creation of a digital object from another one following a deterministic algorithm as Formal Derivations. Formal Derivation represents the transition from an immaterial object to another immaterial object. The resulting instance of digital

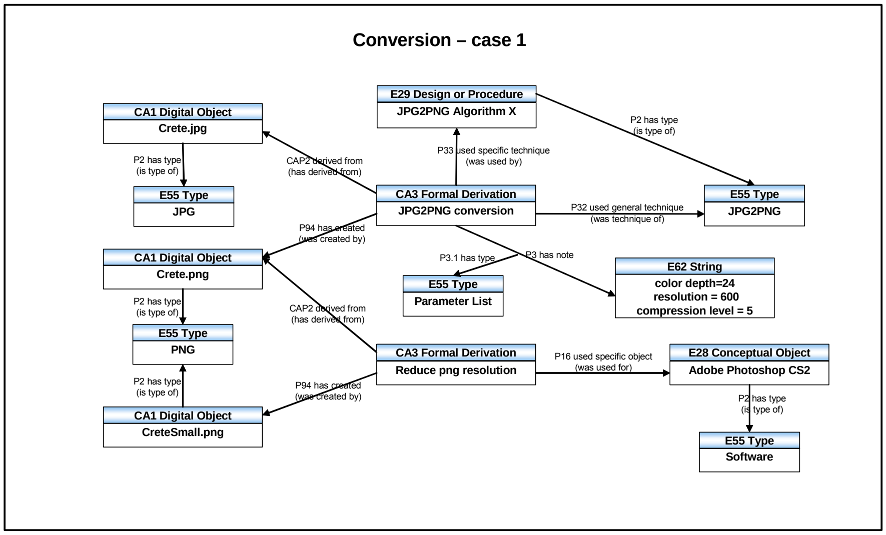


Figure 9: JPG2PNG Converter

object shares representative properties with the original object and can be mechanically reproduced.Suppose we have a converter called **JPG2PNG** and consider three photographs **Crete.jpg**, **Crete.png** and **CreteSmall.png**. The latter two derived from the first photograph by using the converter. **CreteSmall.png** has lower resolution. Figure 9 and Figure 10 illustrates how the above scenario can be modeled using CIDOC CRM classes. In the first case the converter is used to produce **Crete.png** from **Crete.jpg** and then Adobe Photoshop application is used to reduce the resolution of Crete.png and produce **CreteSmall.png**. In the second case both **Crete.png** and **CreteSmall.png** are produced from **CreteSmall.jpg** by using the converter with different parameters.

In both cases, the photographs are instances of **C1 Digital Object**. The class **E55 Type** is used to denote the format of each photograph (see Figure 9) while classes **E54 Dimension** and **E60 Number** are used to model the resolution of each photograph. In general these classes may be used in order to model digital object parameters and their respective values.

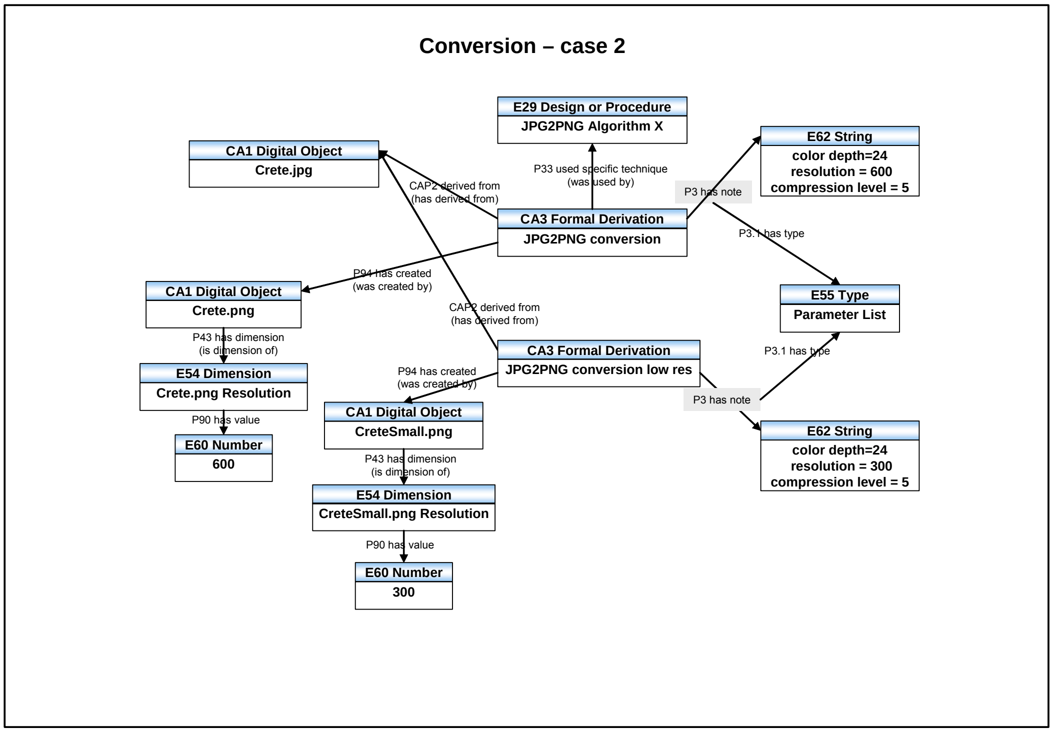


Figure 10: JPG2PNG Converter

The conversion event is an instance of **C3 Formal Derivation** which through the link ***P33 used specific technique (was used by)*** points to the specific algorithm used for the conversion (instance of class **E29 Design** or **Procedure**). In this example the parameters with which the converter was called are not modeled as separate entities but are implied in the name of the **E29 Design or Procedure** instance. Since **E29 Design or Procedure** can be linked with other **E29 Design or Procedure** through ***P69 is associated with***, we can associate the specific call of a converter with the generic converter. **C3 Formal Derivation** is linked to **E55 Type** through ***P32 used general technique (was technique of)*** and denotes the generic type of the conversion.
In Figure 10 we can see how the different parameter list is modeled through the use of property ***P3 has note*** that points to an instance of class **E62 String**.

5.4. Emulation Example

Another example of formal Derivation is the *emulation*. In Figure 11 we present an example of modeling the Handy emulator. To play Atari Lynx games on a Windows-based PC by using the Handy emulator the file LYNXBOOT.IMG is needed as well as Lynx game ROMs. A specific instance of the Handy emulator is an instance of class **E29 Design or Procedure**. The emulation activity is an instance of **C3 Formal**

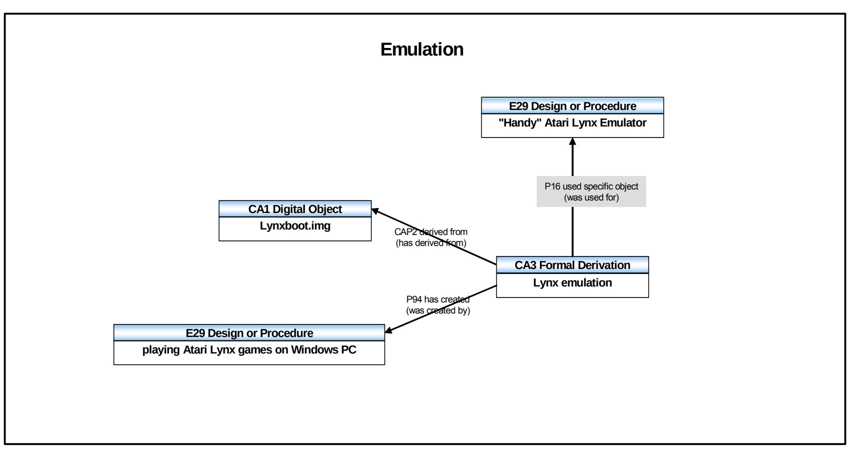
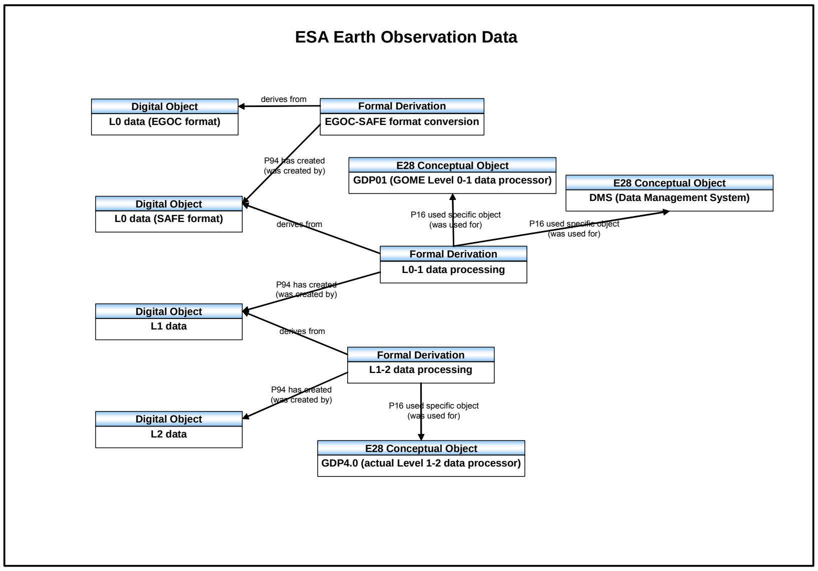


Figure 11: Emulation

Derivation which has the property ***S2 used as source*** pointing to the file LYNXBOOT.IMG which is an instance of **C1 Digital Object.** The result of the emulation is an instance of E29 Design or Procedure.

5.5. Examples with ESA Earth Observation Data

The transformation of L0→ L1 products of ESA.



6. Implementation using Semantic Web Technologies

The purpose of this section is to describe how from an ontology (like CIDOC CRM, FRBRoo) one can define a domain-specific schema (in the form of a Semantic Web ontology) and then use it for documenting the objects of interest. The major part of CIDOC CRM can be straightforwardly represented in Semantic Web languages and such artifacts are already available. However, CIDOC CRM Ontology has nine cases of attributes that start from other attributes (instead of starting from classes). This modeling construct is not straightforwardly supported by Semantic Web languages (and systems). However, all these nine attributes aim at capturing **type** information, therefore they could/should be expressed as elements in the domain specific schemas. This is clarified by the following example.

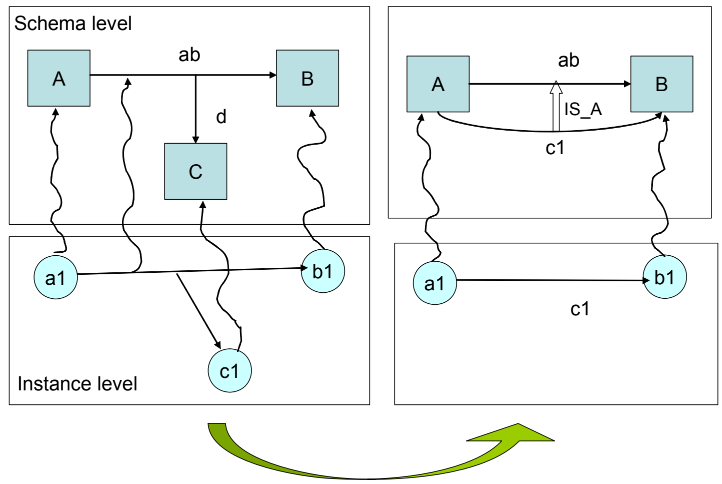


Figure 6-1 Links from Links

The left part of Figure 6-1 shows a conceptual diagram illustrating a part of an ontology plus an instantiation of it. In particular, there is a property **ab** having domain the class **A** and range the class **B**. There is a property **d** whose domain is the property **ab**, having range the class **C**. The figure shows also an instantiation of this schema, specifically **a1** is an instance of class **A**, **b1** is an instance of class **B**, and **c1** is an instance of class **C.**
The above logical structure should be implemented as the right side of Figure 6-1. Specifically, we add at our schema (preferable at the domain specific schema) another property named **ci** which is defined as subproperty (i.e. specialization) of the **ab** property.
The definition of **c1** can be placed at the domain specific schema as in Figure 6-2.

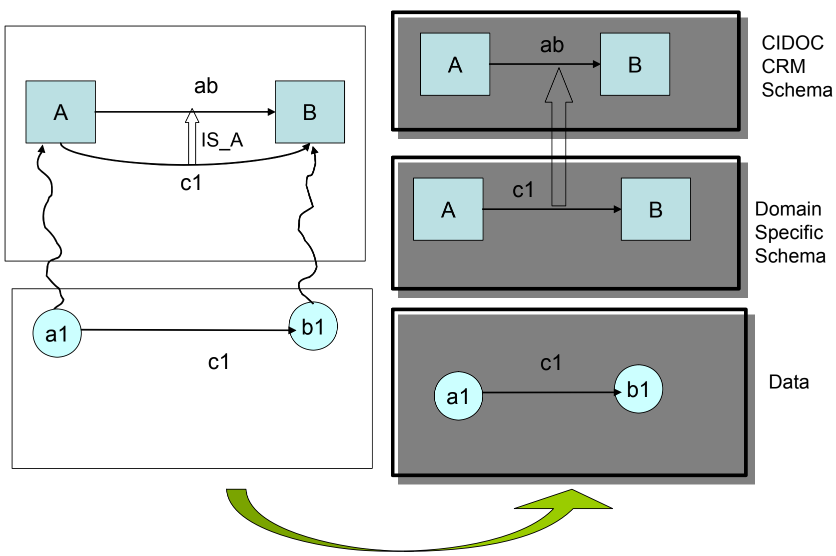


Figure 6-2 Partitioning the knowledge into 3 artifacts

EXAMPLE (may should be placed later on)
For example, in the "Avis de tempete" instantiation of CRM-CIDOC we have the P.14.1 property "in_the_role_of". We can model this by creating a new property named "Composer" as shown in Figure 6-3.

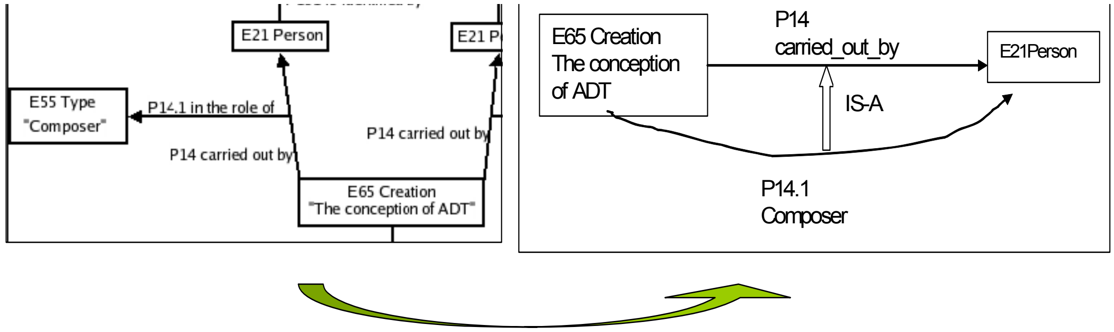


Figure 6-3 Example of Modeling Composer

6.1. Provenance Queries in RQL (Incomplete Draft)

The expression of some of the query templates in RQL[RQL] is given and discussed in the following Table. Below we assume that there is a resource with identity **&myObject.**

#	Descr	Query Template in RQL
1	**Get Creator of a Digital Object**	*Input* : A Digital Object Instance of E28 Conceptual Object *Output*: Instances of E82 Actor Appellation *Description*: E28 Conceptual Object→P94B was created by→E65 Creation→ P14F carried out by(→P14.1 in the role of→E55 Type=Developer)→ E39 Actor→P131F is identified by→E82 Actor Appellation

	RQL	select X5 from
		{X1;E28_Conceptual_Object}P94B_was_created_by{X2;E65_Creation},
		{X2;E65_Creation}P14F_carried_out_by{X3;E39_Actor},
		{X2;E65_Creation}P14_1F_Developper{X3;E39_Actor},
		{X3;E39_Actor}P131F_is_identified_by{X5;E82_Actor_Appellation}
		where X1='**&myObject**'
		
		Note: Instead of **P14_1F_Developper** one could use one of the following:
		{
		PY1_composer,
		PY3_commissioner,
		PY2_writer
		}
		all of them being subproperties of **P14F_carried_out_by** (and having the same domain with the property **P14_1F_Developper)**.

2	**Get the Scanner used to get a Digital Object**	*Input*: A Digital Object Instance of E73 Information Object (Digital Image) *Output*: Instances of E70 Thing Description: E73 Information Object→P39B was measured by→ Digitization Process→P16F used specific object→ (P16.1mode of use→E55 Type=Scanner)→E70 Thing
	RQL	select X3 from {X1;E73_Information_Object}P39B_was_measured_by{X2;C2_Digitization_Process}, {X2;C2_Digitization_Process}P16F_used_specific_object{X3;E70_Thing}, {X2;C2_Digitization_Process}P16_1F_Scanner{X3;E70_Thing} where X1='**&myObject**'
3	**Get the Resolution of a Digital Object**	*Input*: A Digital Object Instance of E73 Information Object (Digital Image) *Output*: Instances of E60 Number *Description*: E73 Information Object→P39B was measured by→ Digitization Process→P40F observed dimension→ E54 Dimension→P90F has value→E60 Number
	RQL	select X4 from {X1;E73_Information_Object}P39B_was_measured_by{X2;C2_Digitization_Process}, {X2;C2_Digitization_Process}P40F_observed_dimension{X3;E54_Dimension}, {X3;E54_Dimension}P90F_has_value{X4;Literal}

		where X1='**&myObject'**
4	**Get the Master Version Of a Digital Object**	*Input*: A Digital Object Instance of E73 Information Object (Digital Image) *Output*: Instance of E18 Physical Thing *Description:* E73 Information Object→P94B was created by→ Digitization Process→S1F_digitized→E18 Physical Thing
	RQL	select X3 from {X1;E73_Information_Object}P94B_was_created_by{X2;C2_Digitization_Process}, {X2;C2_Digitization_Process}S1F_digitized{X3;E18_Physical_Thing} where X1 = **&myObj'**
5	**Get Earlier Versions of**	*Input*: A Digital Derivative Instance of E29 Design or Procedure *Output*: Instance of E70 Thing *Description:* {E29 Design or Procedure→P94B was created by→E65 Creation→ P15F was influenced by→E29 Design or Procedure}* repeat until P15F was influenced by is null
	RQL	select X3 from {X1;E29_Design_or_Procedure}P94B_was_created_by{X2;E65_Creation}, {X2;E65_Creation}P15F_was_influenced_by{X3;E29_Design_or_Procedure} where X1='**&myObj'** *Note*: The above query returns the immediate earlier version(s) of &**myObj**. To get the more earlier version(s), one has to apply the same query again with only difference that instead of "where X1='**&myObj'**" he should write "where X1 In **Z**" where **Z** is the result of the previous query. To get all earlier versions we continue in this way until we get an empty result.
6A	**Get the owner of an object**	*Input*: A physical thing *Output*: The actor that currently owns that thing
	RQL	select X2 from X1;E18_Physical_Thing}P50F_has_current_keeper{X2;E39_Actor} where X1='**&myObj'**
6B	**Get the previous owner of an object**	*Input*: An actor &actor1 and a physical thing &thing1 *Output*: The actor that owned &thing1 just before &actor1
	RQL	select X4 from {X1;E18_Physical_Thing}P50F_has_current_keeper{X2;E39_Actor}, {X2;E39_Actor}P29B_received_custody_through{X3;E10_Transfer_of_Custody}, {X3;E10_Transfer_of_Custody}P28F_custody_surrendered_by{X4;E39_Actor} where X1='**&thing1'** and X2='**&actor1'**

Table 3 Provenance Query Templates in RQL

7. Related Work

Comparison with Open Provenance Model (draft):

Below we compare the proposed approach with OPM [OPM]. The ontology assumed by OPM is very minimalistic and seems not adequate for representing provenance in complex cases. Specifically the ontology assumed by OPM comprises only 3 classes (Artifact, Process, Agent) and 5 associations among them (see Figure 1 of [OPM]). This means, from the perspective of representation adequacy, we could say that provenance information recorded according to the CIDOC CRM extension can be mapped to an OPM-based view but not the other way around.
In addition the assumed ontology by OPM does not explicitly contain the concept of Event. The notion of event is of prominent importance, not only because it allows tracing the history of an object but it also aids the integration of all information that concerns an object. However the way OPM interprets Processes resemble events (however the ontological structure is not rich).
OPM also proposes a number of inference rules. Some of the inference rules of OPM are equivalent to the inferences due to the subclassOf relationships of CIDOC CRM extension. Some other could be expressed over the CIDOC CRM ontology by adopting an appropriate Rule Language.

The expression of CIDOC CRM extension in Semantic Web (SW) languages offers standard formats for exchanging provenance data (i.e. RDF/XML, Trig) while the SW data management tools can be used for storing and declaratively querying and updating such repositories. In addition, inferences could be defined by adopting SWRL [SWRL] rules.

8. Concluding Remarks

This paper described an extension of the CIDOC CRM ontology (ISO 21127:2006) able to capture the modeling and the query requirements regarding the provenance of digital objects. We discussed the relationship with OAIS (ISO 14721:2003) and provided a number of indicative examples of modeling for provenance and related query templates. We described the expression in RDF(/S) and provided query templates in RQL.

Acknowledgements

This work was partially supported by the EU project CASPAR (FP6-2005-IST-033572).

9. References

- [Crofts et al. 2007] Nick Crofts, Martin Doerr, Tony Gill, Stephen Stead, Matthew Stiff (editors), **Definition of the CIDOC Conceptual Reference Model**, October 2006, version 4.2.1 of the reference document. http://cidoc.ics.forth.gr/official_release_cidoc.html

- [OAIS] Consultative Committee for Space Data Systems, **Reference Model for an Open Archival Information System (OAIS)**, CCSDS 650.0-B-1 Blue Book, January 2002, http://public.ccsds.org/publications/archive/650x0b1.pdf

- [OPM] **The Open Provenance Model**, Moreau, L. and Freire, J. and Myers, J. and Futrelle, J. and Paulson, P.,University of Southampton, 2007, }

- [RQL] G. Karvounarakis, V. Christophides and D. Plexousakis, **RQL: A Declarative Query Language for RDF**,11th World Wide Web Conference (WWW, Hawaii, USA, May, 2002

- [StarLion] Y. Tzitzikas and D. Kotzinos, and Y. Theoharis, **On Ranking RDF Schema Elements (and its Application in Visualization**, Journal of Universal Computer Science, 13(12), 2007

- [SWRL] SWRL: A Semantic Web Rule Language Combining OWL and RuleML, W3C, http://www.w3.org/Submission/SWRL/

- [Tzitzikas, 2007a] Y. Tzitzikas: **Dependency Management for the Preservation of Digital Information.** 18th International Conference on Database and Expert Systems Applications, DEXA'2007, Regensburg, Germany, September 2007

- [Tzitzikas, 2007b] Y. Tzitzikas, G. Flouris: **Mind the (Intelligibility) Gap**, 11th European Conference on Research and Advanced Technology for Digital Libraries, ECDL'2007, Budapest, Hungary, September 2007

- [Tzitzikas, 2007c] Y. Tzitzikas, **On Preserving the Intelligibility of Digital Objects though Dependency Management**, International Conference PV'2007 (Ensuring the Long-Term Preservation and Value Adding to Scientific and Technical Data), Oberpfaffenhofen/Munich, Germany, October 2007

APPENDIX A CIDOC CRM Extensions in RDFS

```
<rdf:RDF xml:lang="en" xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
       xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
       xmlns:cidoc="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#">

<rdfs:Class rdf:ID="C1_Digital_Object">
       <rdfs:comment></rdfs:comment>
       <rdfs:subClassOf
rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E54_Dimension"/>
       <rdfs:subClassOf
rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E73_Information_Object"/>
</rdfs:Class>
<rdfs:Class rdf:ID="C2_Digitization_Process">
       <rdfs:comment></rdfs:comment>
       <rdfs:subClassOf
rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E11_Modification"/>
       <rdfs:subClassOf
rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E16_Measurement"/>
       <rdfs:subClassOf
rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E65_Creation"/>
</rdfs:Class>
<rdfs:Class rdf:ID="C3_Formal_Derivation">
       <rdfs:comment></rdfs:comment>
       <rdfs:subClassOf
rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E65_Creation"/>
</rdfs:Class>
<rdfs:Class rdf:ID="C4_Norm">
       <rdfs:comment></rdfs:comment>
       <rdfs:subClassOf
rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E28_Conceptual_Object"/>
</rdfs:Class>
<rdfs:Class rdf:ID="C5_Copyright">
       <rdfs:comment></rdfs:comment>
       <rdfs:subClassOf
rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E30_Right"/>
</rdfs:Class>
<rdfs:Class rdf:ID="C6_Copying">
       <rdfs:comment></rdfs:comment>
       <rdfs:subClassOf
rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E7_Activity"/>
</rdfs:Class>
<rdf:Property rdf:ID="S1F_digitized">
       <rdfs:domain rdf:resource="#C2_Digitization_Process"/>
       <rdfs:range
rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E18_Physical_Thing"/>
</rdf:Property>
<rdf:Property rdf:ID="S1B_was_digitized_by">
       <rdfs:domain
rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E18_Physical_Thing"/>
       <rdfs:range rdf:resource="#C2_Digitization_Process"/>
</rdf:Property>
<rdf:Property rdf:ID="S2F_used_as_source">
       <rdfs:domain rdf:resource="#C3_Formal_Derivation"/>
       <rdfs:range
rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E70_Thing"/>
</rdf:Property>
<rdf:Property rdf:ID="S2B_was_source_for">
       <rdfs:domain
rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E70_Thing"/>
       <rdfs:range rdf:resource="#C3_Formal_Derivation"/>
</rdf:Property>
<rdf:Property rdf:ID="S3F_allows">
       <rdfs:domain
rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E30_Right"/>
       <rdfs:range
rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E7_Activity"/>
</rdf:Property>
<rdf:Property rdf:ID="S3B_is_allowed_by">
       <rdfs:domain
rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E7_Activity"/>
       <rdfs:range
rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E30_Right"/>
</rdf:Property>
<rdf:Property rdf:ID="S4F_violates">
       <rdfs:domain
rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E7_Activity"/>
       <rdfs:range
rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E30_Right"/>
</rdf:Property>
<rdf:Property rdf:ID="S4B_is_violated_by">
       <rdfs:domain
rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E30_Right"/>
       <rdfs:range
rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E7_Activity"/>
</rdf:Property>
<rdf:Property rdf:ID="S5F_makes_use_of">
       <rdfs:domain
rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E7_Activity"/>
       <rdfs:range
rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E30_Right"/>
```

```
</rdf:Property>
<rdf:Property rdf:ID="S5B_is_used_by">
        <rdfs:domain
rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E30_Right"/>
        <rdfs:range
rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E7_Activity"/>
</rdf:Property>
<rdf:Property rdf:ID="S6F_holds">
        <rdfs:domain
rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E39_Actor"/>
        <rdfs:range
rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E30_Right"/>
</rdf:Property>
<rdf:Property rdf:ID="S6B_is_granted_to">
        <rdfs:domain
rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E30_Right"/>
        <rdfs:range
rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E39_Actor"/>
</rdf:Property>
<rdf:Property rdf:ID="S8F_copies_to">
        <rdfs:domain rdf:resource="#C6_Copying"/>
        <rdfs:range
rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E73_Information_Object"/>
</rdf:Property>
<rdf:Property rdf:ID="S8B_is_created_by">
        <rdfs:domain
rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E73_Information_Object"/>
        <rdfs:range rdf:resource="#C6_Copying"/>
</rdf:Property>
<rdf:Property rdf:ID="S9F_has_validity">
        <rdfs:domain rdf:resource="#C4_Norm"/>
        <rdfs:range rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E52_Time-
Span"/>
</rdf:Property>
<rdf:Property rdf:ID="S9B_is_validation_period_of">
        <rdfs:domain
rdf:resource="http://cidoc.ics.forth.gr/rdfs/caspar/cidoc.rdfs#E52_Time-Span"/>
        <rdfs:range rdf:resource="#C4_Norm"/>
</rdf:Property>
</rdf:RDF>
```

APPENDIX B CIDOC CRM Extensions

C1 Digital Object

Subclass of: E73 Information Object
E54 Dimension

Scope note: This class comprises identifiable immaterial items, that are composed of a set of bit sequences, such as data sets, e-texts, images, audio or video items, software, etc., that have an objectively recognizable structure and are documented as single units. A C1 Digital Object can represent things at any arbitrary level of granularity. This means that, for instance, it can represent a journal, an individual issue of a journal, an individual article in the journal, or a single table in that article.

A C1 Digital Object does not depend on a specific physical carrier, and it can exist on one or more carriers simultaneously.

Examples:

- image BM000038850.JPG from the Clayton Herbarium in London
- texas_flood_21-28june07_graph.gif

Properties:

C2 Digitization Process

Subclass of: E11 Modification
E16 Measurement
E65 Creation

Superclass of:

Scope note: This class comprises events that result in the creation of instances of C1 Digital Object that represent the appearance and/or form of an instance of E18 Physical Thing such as paper documents, archaeological objects, films, photographs, etc.
This class represents the transition from a material thing to an immaterial representation of it. The characteristic subsequent processing steps on digital objects are regarded as instances of C3 Formal Derivation.

Examples:

- the scanning of the performance handbook of Avis de Tempête
- the digital photographing of van Gogh's self portrait
- the audio visual recording of the 17-11-2004 performance of ADT at the Opéra de Lille

Properties:

S1 digitized (was digitized by): E18 Physical Thing

C3 Formal Derivation

Subclass of: E65 Creation

Superclass of:

Scope note: This class comprises events that result in the creation of a C1 Digital Object from another one following a deterministic algorithm. The resulting instance of digital object shares representative properties with the original object and can be mechanically reproduced.
This class represents the transition from an immaterial object to another immaterial object.

Examples:

- the reduction of the resolution of image BM000038850.JPG from the Clayton Herbarium in London to 300dpi

Properties:

S2 used as source (was source for): E70 Thing

APPENDIX C

One rising question is how CIDOC CRM is related to OAIS RepInfo and with the dependency-based intelligibility services described in [Tzitzikas 2007a/b/c]. The latter approach is a based on a minimal core ontology whose structure is shown in Figure 4. For more refer to [Tzitzikas 2007a/b/c].

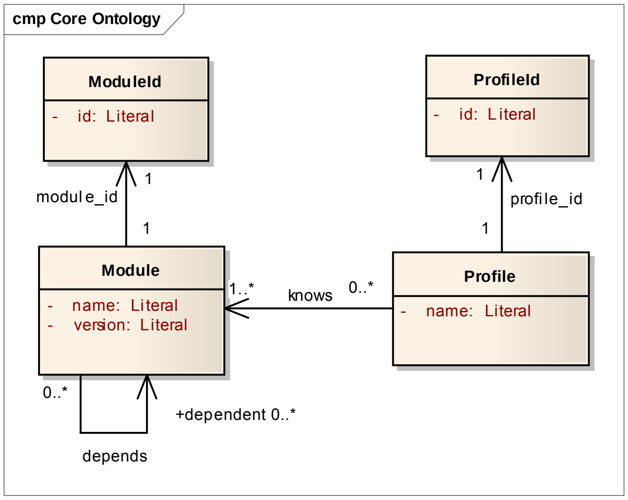


Figure 4 The minimal code ontology for dependencies

The class **Module** may be mapped to the entity **E73 Information Object** or with **C1 Digital Object**. The latter is the newly added entity of CIDOC CRM, defined as a subclass of **E73 Information Object** and of **E54 Dimension**.
The relation ***depends*** (between Modules) may be mapped to the property **P2 has Type.** Recall the example given in Figure 10 (with Crete.jpg, Crete.png, CreteSmall.png) all defined as instances of the class **C1 Digital Object**. Notice the use of the **P2 has type** relation.

In the general case, one may find out that a intelligibility dependency corresponds to certain relationships, or paths of relationships, over an existing conceptual model. Then he could either "merge appropriately" the schema of his conceptual model with that dependencies. For instance if one wants every CIDOC CRM **E73_Information_Object** to be considered as **Module**, then he/she could define **E73_Information_Object** as a subclass of **Module**. Alternatively, we could either manually or through a declarative update language, classify his data also wrt the dependencies ontology..

In that case the dependency ontology can be considered as the schema of a read-only view of more sophisticated conceptual models. At the technical/implementation level, the ability of multiple-classification and of inheritance (of Sem Web languages) gives this flexibility.