Semantic Web 14 (2023) 553–584
DOI 10.3233/SW-222798
IOS Press

A strategy for archives metadata representation on CIDOC-CRM and knowledge discovery

Dora Melo [a,c,d,*], Irene Pimenta Rodrigues [b,c] and Davide Varagnolo [b]
[a] *Coimbra Business School|ISCAC, Polytechnic of Coimbra, Portugal*
E-mail: dmelo@iscac.pt
[b] *Department of Informatics, University of Évora, Portugal*
E-mails: ipr@uevora.pt, d.varagnolo@studenti.unipi.it
[c] *NOVA Laboratory for Computer Science and Informatics, NOVA LINCS, Portugal*
[d] *Polytechnic of Coimbra, Coimbra Business School Research Center|ISCAC, Coimbra, Portugal*

Editor: Eero Hyvönen, Aalto University and University of Helsinki, Finland
Solicited reviews: Carlo Meghini, CNR-ISTI, Italy; Four anonymous reviewers

Abstract. This paper presents a strategy for the semantic migration of Portuguese National Archives records into CIDOC-CRM standard, an ontology developed for museums, within the context of the EPISA project. The approach to automatically populate the CIDOC-CRM is based on Mapping Description Rules to semantically translate the archives descriptive information into CIDOC-CRM representation. The compliance of the CIDOC-CRM model recommendations guarantees that the populated CIDOC-CRM ontology of archives descriptive information verifies interoperability, and could be linked and integrated with other populated CIDOC-CRM ontologies. In the information modelling, requirements on the mapping representation, due to the intent of interpreting natural language text to automatically extract information of metadata text fields and to interpret natural language queries, are taken into account. To automatically interpret the Mapping Description Rules, `OWL API` was used to obtain the set of assertions that represents the information in the target ontology and two datasets are available with some migration examples. The exploration of the knowledge representation is done through some Description Logic queries to highlight the advantages of having this new representation of the National Archives. The evaluation of the resulting representation can be done automatically proving its correctness for the metadata that has a direct representation in CIDOC-CRM.

Keywords: Knowledge representation and reasoning, natural language processing, archives ontology, semantic migration, CIDOC-CRM, linked data science

1. Introduction

This work is done in the context of the EPISA project (Entity and Property Inference for Semantic Archives), a research project involving the Portuguese National Archives, Torre do Tombo (ANTT), the archival experts from ANTT, and Information and Computer Science researchers. EPISA intends to design a prototype, an open-source

*Corresponding author. E-mail: dmelo@iscac.pt.

knowledge platform, to represent archival information on a linked data model. One of the project major tasks is the semantic migration, i.e, the process to extract and represent the relevant entities and their properties from the existing records in the actual DigitArq [29], the archive national system that uses well-established description standards, namely the ISAD(G) (General International Standard Archival Description) [12] and ISAAR(CPF) (International Standard Archival Authority Record for Corporate Bodies, Persons and Families) [34] with a hierarchical structure adapted to the nature of archival assets.

The data model and description vocabularies adopted are built upon the CIDOC-CRM (Conceptual Reference Model) standard [5], an ontology developed for museums by the International Committee for Documentation (CIDOC) of the International Council of Museums (ICOM) [5,19].

The aim of this paper is to introduce an approach to automatically populate the CIDOC-CRM with the Portuguese National Archives metadata. The methodology is based on Mapping Description Rules to semantically translate the archives descriptive information into CIDOC-CRM ontology representation.

The Mapping Description Rules are a set of rules to formally define the translation of one representation model into another model, in particular, of ISAD(G) into CIDOC-CRM representation. The proposed set of rules are written using the Mapping Description Language proposed in [3]. This language is suitable to express hierarchical models, such as ISAD(G) representation and OWL ontologies. The ISAD(G) model establishes the descriptions of the archival materials, based on the principle of *respect des fonds* within a multi-level description, that defines some elements to constitute the description of an archival entity. The process to translate these elements to an OWL ontology requires defining how each element is represented and how they are connected in the ontology. An archival entity and its elements are represented as instances of ontology classes and are connected by ontology properties. The classes and properties, chosen to represent each archival description, define the mapping rules and the ontology representation model. For instance, in this work, an archival entity is always represented as an instance of 'E_{31} Document' that documents (property 'P_{70} documents') an instance of 'E_{22} Human-Made Object' (representing the physical object).

The Mapping Description Rules, as defined, can be easily adapted to the use of other ontologies.

The compliance of the CIDOC-CRM model recommendations guarantees that the populated CIDOC-CRM ontology of archives descriptive information verifies interoperability, and could be linked and integrated with other populated ontologies using CIDOC-CRM representation.

The semantic web representation of the archival information, which contains such a rigid structure as the one imposed by ISAD(G), will enable to restructure the information in different views, such as a chronological view of the production dates, production geographical places, people to whom the information concerns, etc. Such reorganization is a difficult task in a relational model database, even when using full string search. The semantic web representation enables end users to search the archival metadata using state of the art tools in an efficient way.

The Portuguese National Archives have their databases organized by Regional archives that are not integrated, which implies the search in each database. The semantic web model enables to integrate the information of all databases. Furthermore, it is also possible to integrate information from other archives, national or international, represented in CIDOC-CRM, by using the name of known entities to link data. In addition, the information extracted from the text elements allows to represent new information, such as births, incorporations or transfers of documents, that is not represented in the relational model of the DigitArq database.

The remainder of this paper is divided into the following sections. Section 2 presents the norms and formats to universally describe archives metadata, proposals for mapping ISAD(G) into ontologies such as CIDOC-CRM, natural language interpretation of queries and raw text to automatically populate an ontology, and a brief summary of current work related with CIDOC-CRM representation and interfaces to query OWL2 knowledge base.

The representation of ISAD(G) and ISAAR(CPF) Archives Metadata in CIDOC-CRM is presented in Section 3. This section introduces the methodology based on Mapping Description Rules for automatizing the migration process, presents the CIDOC-CRM recommendations for modelling information, to guarantee the effectiveness and the consistency of the final populated ontology, as well as some requirements on the mapping representation due to the intent of interpreting natural language text to automatically extract information of metadata text fields and to interpret natural language queries.

Section 4 presents the architecture of the migration process from DigitArq HTML records into CIDOC-CRM and describes in detail each one of its steps. Some illustrative examples are presented for clarification and better

understanding. A discussion about the evaluation of the migration process and the migration to other ontologies is also presented.

The exploration of the knowledge base, as a result of the migration process, is described in Section 5. A set of questions performed over the knowledge base is presented in order to confirm that the CIDOC-CRM Ontology representation of the DigitArq metadata is correct, to explore the information extracted from the texts and also to explore new ways of organizing the information. To help and facilitate the task of querying the knowledge base, an application program interface was also developed and it is also presented in this section.

In Section 6, a set of open problems that arose from occurred issues while developing and implementing the Mapping Description Rules, together with the analysis of different examples, is discussed.

The conclusions, as well as further work and a future evaluation are drawn in Section 7.

Finally, in the Appendix, the Mapping Description Rules to represent DigitArq information in CIDOC-CRM are presented.

2. The archival description scenario

The International Council of Archives[1] (ICA) defines archives as "the documentary by-product of human activity retained for their long-term value." They are characterized as contemporary records created by individuals and organisations about their business, providing information on past events. These records can be of a wide range of formats including written, photographic, films, sound, digital and analogue.

The aim of the ICA is to promote the management of archives, and the preservation of the archival heritage of humanity around the world. The sharing of experiences, research and ideas on professional archival, records management, as well as on the management and organisation of archival institutions, are part of their strategy.

In this follow-up, the ICA Committee Description Standards developed the General International Standard Archival Description (ISAD(G)) [12], which provides general guidance for creating descriptions of archival materials, establishing a model based on the principle of *respect des fonds* within a multi-level description. ISAD(G) defines 26 elements that may be combined in seven areas to constitute the description of an archival entity. These areas and provide general content guidelines and are identified by Identity Statement, Context, Content and Structure, Condition of Access and Use, Allied Materials, Note, and Description Control. The structure and content of the information in each of those elements should be formulated in accordance with applicable national rules. As general rules, these are intended to be broadly applicable to descriptions of archives regardless of the nature or extent of the unit of description (subsequently identified also as just 'unit').

The International Standard Archival Authority Record for Corporate Bodies, Persons and Families (ISAAR (CPF)) [34], also developed by the same ICA Committee Descriptions Standards, provides guidance for preparing archival authority records which introduce descriptions of entities, such as corporate bodies, persons and families, associated with the creation and maintenance of archives.

The ISAD(G) content model, along with the ISAAR (CPF), serves as the basis for the development of the guidance document for the standardization of Portuguese archival descriptions [27]. This document was developed by the General Directorate for Book, Archives, and Libraries (DGLAB),[2] through an archival description standardization working group. Besides the introduction of the ISAD(G) elements and ISAAR (CPF) descriptions, the Portuguese guidance has two main purposes: first, the inclusion of the detailed perspective of the lower levels of description, such as installation unit, compound document and simple document (named as Item); and second, the addition of a unifying view of the description that included coherent description of documents in electronic form.

The need for a means to facilitate the archivists work, as well as coherent finding aids to help users and archivists attain the artefacts they seek, were the main reasons for the development of the DigitArq [29] platform. DigitArq is

[1] https://www.ica.org/en

[2] The DGLAB (http://dglab.gov.pt/) is a public body under the Portuguese Ministry of Culture's responsibility, is a central service of the direct administration of the State, endowed with administrative autonomy, whose mission is to ensure the coordination of the national archives system and the implementation of an integrated policy for non-school books, libraries and reading.

characterized by a common digital format based on an international standard and an archival management software to maintain all information, supported by a centralised repository to store all the collected material.

For Archives information representation, the RIC-O (Record In Context Ontology) [28] and CIDOC-CRM [5] ontologies were considered. The RIC-O model is an OWL Ontology for describing archival record resources and their contextual entities, which has an application converter of the ICA Records from French National Archives (ANF) in Contexts standard (ICA RiC).[3] However, this model is still in development and not so well-established as CIDOC-CRM, which is a model widely used in the Heritage domain, and based on well-documented experiments of modeling Museums, Archaeology, and Architecture domains, and also because some of EPISA Project Team members, the Archival experts from DGLAB, were already involved with the CIDOC-CRM representation, motivated the CIDOC-CRM as first choice. Additionally, semantic integration and interoperability can be achieved through the use of CIDOC-CRM, since there are many platforms available to access the information in CIDOC-CRM representation for several domains.

The development of the semantic migration process of the DigitArq metadata uses the CIDOC-CRM ontology as a data model and description vocabulary. The semantic mapping of archival metadata into the CIDOC-CRM Ontology can be straightforward for some elements [3,10,14].

The first approach to present a set of mapping rules was a study to explore the representation expressiveness of CIDOC-CRM into archival metadata domain [32]. This approach presents a set of rules which allows to map Encoded Archival Description (EAD) into CIDOC-CRM representation. EAD is a XML language designed to represent the ISAD(G) elements in XML syntax and is maintained by the standards initiative of the Library of Congress, and a rigorous mapping between EAD and ISAD(G) and vice-versa are maintained [8]. More recent, this first study was extended with a set of mapping rules and a language to write them [3]. Using this mapping rules, a conceptual ontology for Archival Knowledge Model was proposed in [10], with the purpose of querying archival or historical knowledge bases, where natural language queries are translated to the CIDOC-CRM and appropriate extensions.

The semantic integration of CIDOC-CRM with other standards has been a recurring goal [7]. An example of an effort in this regard is the proposal for semantic integration of collection description illustrated with Dublin Core and CIDOC-CRM [14].

The importance of the migration process lies not only in the direct translation of the ISAD(G) elements, but also in the possibility of adding information to the knowledge base that can be extracted and inferred from the textual elements. In fact, there are elements of ISAD(G) descriptions whose content is free text about the record itself and for which there are no general mapping rules available. This content must be interpreted in the CIDOC-CRM ontology context in order to represent the entities, events, locals, dates, relations and properties in the ontology. This process is achieved by applying Natural Languages Processing (NLP) techniques. OntoPrima [15] is a NLP-based Ontology Population system that extracts instances of concepts and instances of relations from text, to populate a given ontology based on NLP techniques for language processing, semantic web techniques (RDFS, RDF, Jena APIs) for knowledge modeling and representation, and on domain expert's intervention to validate extracted instances. This topic is explored in other works such as [6,16,18].

In the past few years, some interfaces were developed for CIDOC-CRM knowledge bases, mainly in the cultural heritage domain, such as OpenArcheo [17], that allows the users to create complex query with an user's friendly GUI and facilitates the task of searching for information that users seek to find, or even Arches heritage inventory and management system [26] and ONTOME a collaborative ontology management environment [1,2]. An example of a differentiation tool is the interface for manipulating narratives, Narrative Building and Visualisation Tool [25], that allows the users to add new narratives and visualizes information about them. All these platforms are a mean to integrate different domain knowledge bases for interoperability.

[3] https://blog-ica.org/2020/06/13/ric-o-converter-an-example-of-practical-application-of-the-ica-records-in-contexts-standard-ica-ric/

3. Representing ISAD(G) and ISAAR(CPF) archives metadata in CIDOC-CRM

As mentioned before, ISAD(G) content model is based on the principle of *respect des fonds* within a multi-level description. This principle has as practical consequence that archival description proceeds from the general to the specific, in order to represent the context and hierarchical structure of the fonds and its component parts. Generically, this means that each level of description can be subdivided into the sub-levels considered necessary to mirror the different documentary realities. In addition, the multi-level description model also complies with the following rules: set information relevant to the level of description, with the aim of accurately representing the context and content of the unit of description; the existence of a link between descriptions, in order to make explicit the position of the unit of description in the hierarchy; and no repetition of information, in order to avoid redundancy of information in hierarchically related archival descriptions. Figure 1 presents the model of the levels of a possible arrangement of a fonds. Archival materials are organized in fonds, top of the hierarchy. Each fonds can be composed of other archival materials organized in the sublevels series, subseries, files, or items. The organization of the fonds in the intermediate levels need to comply with the hierarchy, even though some levels are missing.

Concerning to each unit at some level of description, all 26 information elements provided for in ISAD(G) can be considered, in their entirely, at any level of description, according to the desired degree of completeness. However, just the following elements are considered essential for international exchange of descriptive information [12,27]: reference code; title; creator; date(s); extent of the unit of description; and level of description. The reference code is the information that allows to identify uniquely the unit and to provide a link to the description that represents it. The creator of the unit identifies the corporate body, family or person that created, accumulated and/or maintained records in the conduct of personal or corporate activity. The date(s) identifies and records the date(s) of the unit, such as date range or creation date. The extent of the unit is the information that allows to identify and describe the physical or logical extent and the medium of the unit of description. Finally, the level of description is the position of the unit in the hierarchy of the fonds (Fig. 1).

Taking this knowledge into account, it was necessary to establish the principles of information representation to ensure that the migration process of the archives' metadata into the CIDOC-CRM ontology is successfully com-

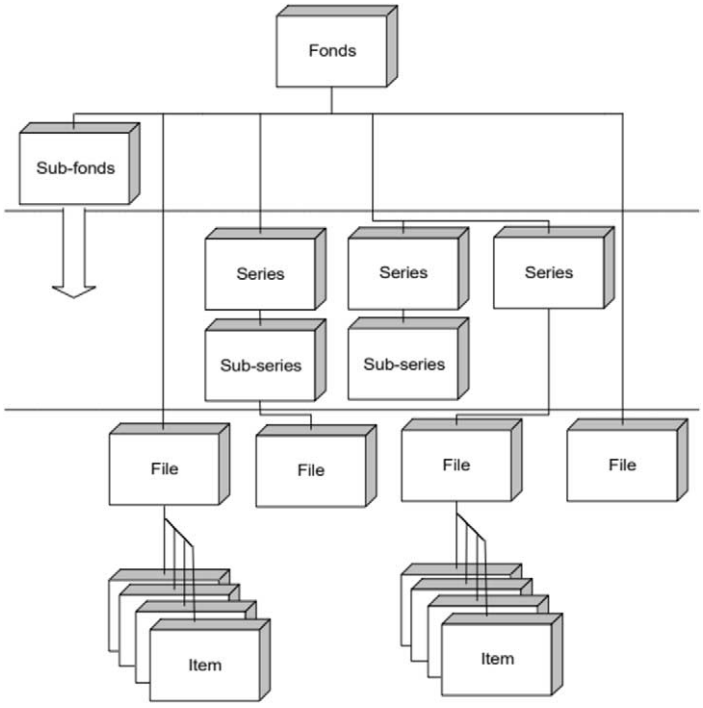


Fig. 1. Model of the levels of arrangement of a fonds [12, p. 36].

pleted. Therefore, the representation of the archives' metadata in CIDOC-CRM uses the criteria explained in the following subsections.

3.1. CIDOC-CRM recomendations

The translation of the Archival metadata into the CIDOC-CRM representation follows the main principles of the CIDOC-CRM model[4] [5,31].

1. The introduction of a new class should comply with the minimality modelling principle of CIDOC-CRM:
   "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".
   Regarding properties, a new one only should be added if "it is a key concept in the practical scope".
2. The representation of terms that declare that an object belongs to a particular category of items follows the CIDOC-CRM specific modeling constructs 'about types'.
   The class 'E_{55} Type' comprises such terms from thesauri and controlled vocabularies used to characterize and classify instances of CIDOC-CRM classes. Therefore, instances of 'E_{55} Type' represent concepts (universals) in contrast to instances of 'E_{41} Appellation', which are used to name instances of CIDOC-CRM classes.
   In addition, the property 'P_2 has type (is type of)' provides the mechanism to specialize the classification of CIDOC-CRM instances to any level of detail, by linking to external vocabulary sources, thesauri, classification schemas or ontologies.
3. The cases in which categorization is established in the relationship (property) between two individuals, i.e., stating the role of a relation between individuals, the representation also follows the CIDOC-CRM specific modeling constructs 'about types'.
   With an analogous purpose of the 'P_2 has type (is type of)' property, some properties of the CIDOC-CRM are associated with an additional property and are numbered with a '.1' extension [33]. The range of these properties of properties always falls under 'E_{55} 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 'E_{55} Type'.

3.2. Information useful for natural language interpretation of text or queries

With the focus on the development of an interface to query the knowledge base in Natural Language and, in the near future, the automatic information extraction from text, it is important that the information representation in the ontology facilitates the interpretation process of Natural Language expressions. The interpretation of a Natural Language expression requires the inference of instances, classes or/and properties in the ontology.

Consider, for instance, the concept 'Country' that could be represented in the ontology as a class. If the ontology does not have a class with that concept, a general concept should be considered, such as 'Place', and then define 'Country' as a subclass of 'Place'. However, by the first main principle of CIDOC-CRM recommendation, a new subclass should not be created. Therefore, there are two alternatives, first one is to create an instance of the class 'Place', losing the subconcept 'Country', and the second one is to create an instance of the class 'Place' connected with an instance of 'E55 Type' with value 'Country', using the property 'P2 has type'.

The second alternative is more informative than the first one, since it is possible to model the subconcept 'Country', allowing to retrieve easily all instances that have type 'Country'. Therefore, the following rule is taken into account when defining the mapping between ISAD(G) and CIDOC-CRM:

1. To create an instance V, of a concept C1, if the concept C1 is a subclass of another concept C2 and the ontology does not have C1 but has C2, then the mapping is defined by establishing the concept C1 as a type of an instance V of the concept C2 (see Fig. 2).

[4] CIDOC-CRM version 7 and its RDF Schema expression.

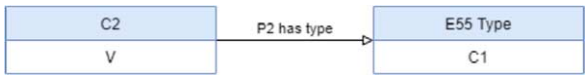


Fig. 2. Representation of an instance V of the class C1, which is a subclass of the class C2.

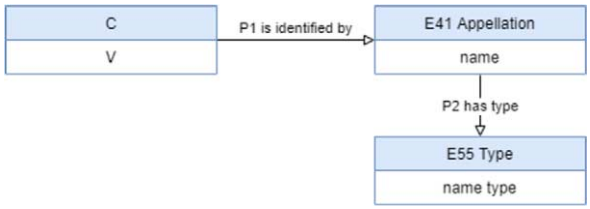


Fig. 3. Representation of the type of an appellation.

This rule is always used whenever the concept C1 can be represented by Natural Language expressions in an ontology query. For instance, consider the query "Which are the countries that produced baptism materials?". The result should be a list of instances of 'Place' that has type 'Country' and produced baptism materials. To interpret the meaning of an expression, such as 'Country', a noun, it is necessary to search for:

- an ontology class with a label similar to the expression "Country" → 'E53 Place'
- an instance of the class 'E55 Type', with value similar to the expression "Country" → 'E53 Place' and 'P2 has type' 'E55 type'{= Country}

The similarity calculus will be higher for the second interpretation, since the expression 'Country' is closer to the second expression than to the expression 'Place'. The second interpretation is enabled by representing the concept 'Country' with the rule 1 above and this representation captures the meaning of the expression 'Country' in the query.

Another rule that is taken into account in the mapping process is that:

2. When creating an instance of the class 'E_{41} Appellation' (or 'E_{42} Identifier'), with value *name*, to identify an instance of a class C, it is necessary to clarify the classification of the role between the *name* and the instance of the class C, by adding a type to the *name* (see Fig. 3).

This rule is always used whenever the *name* can be represented by Natural Language expressions in an ontology query. For instance, consider the query "Which are the institution's abbreviations?". The result should be a list of instances of the class 'E_{41} Appellation' (or 'E_{42} Identifier') that are classified as 'institution abbreviation'.

The matching between the sentences terms (nouns, adjectives, prepositions, verbs, named entities) and classes, properties and instances of an ontology is a common step in natural language interpretation for querying an ontology or mining text to populate an ontology [6,15,21,22].

3.3. Mapping description rules

As mentioned before, each unit, at some level of description, has a well-known structure of information defined by the ISAD(G) elements. In order to define the representation of each unit, the elements can be grouped, according to their content and what they refer to, and associated with three concepts, namely the object itself that belongs to the physical archive; the digital registration that describes the object; and the language properties associated to the object (when they exist). These three concepts are mapped into the following CIDOC-CRM classes, respectively 'E_{22} Human-Made Object'; 'E_{31} Document'; and 'E_{33} Linguistic Object'. Whereby, each unit is itself mapped into an instance of 'E_{31} Document', that documents (property 'P_{70} documents') a physical object (an instance of 'E_{22} Human-Made Object) and refers to (property 'P_{67} refers to') a conceptual object (an instance of 'E_{33} Linguistic Object').

The hierarchical structure of the archives is represented using the relation 'P_{106} is composed of' between the 'E_{31} Document' and their sub-documents (also represented as individuals of the class 'E_{31} Document').

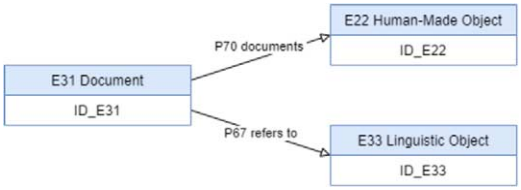


Fig. 4. Rule No. 1 – Unit of description.

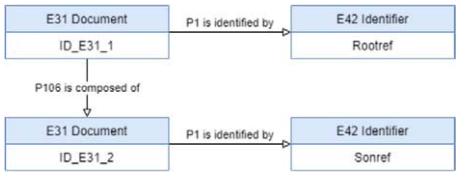


Fig. 5. Rule No. 17 – Hierarchy.

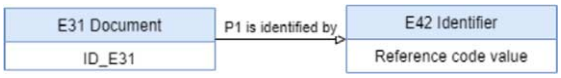


Fig. 6. Reference code representation as the unique identifier of a document.

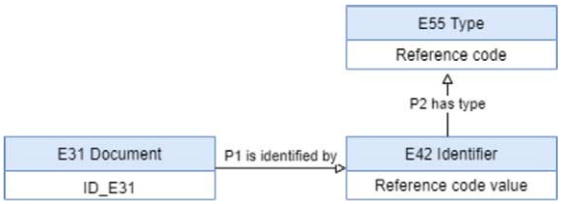


Fig. 7. Reference code representation as the unique identifier of a document and a type.

These representations follow the CIDOC-CRM recommendations, and similar approaches for representing archives and collections, presented in [3,14,32]. The representation of the archival description units in the CIDOC-CRM Ontology is done through rules that express the metadata mapping into the ontology entities. These rules define the set of Mapping Description Rules that establishes the basis for the automatic migration process. Table 5, in the Appendix, presents some of the Mapping Description Rules defined.

Therefore, the representation of the unit explained before is translated into the rule No. 1, showed in Fig. 4, and the hierarchy of the archive is captured in rule No. 17, and showed in Fig. 5.

Consider, for instance, the ISAD(G) element 'Reference code', with value 'PT/TT/...'. Each unit is uniquely identified by this code. The 'Reference code' can be represented as an instance of the class 'E_{42} Identifier', and to represent that the value of the reference code is the identification of the unit, it is possible to use the property 'P_1 is identified by', resulting the statement 'E_{31} Document' 'P_1 is identified by' 'E_{42} Identifier {= PT/TT/...}', see Fig. 6.

However, in this representation, the information that the 'Reference code' is the identifier of the document is implicit. If this information needs to be explicit, then it is possible to apply a type to the identifier with the rule 'E_{42} Identifier {= PT/TT/...}' 'P_2 has type' 'E_{55} Type {= Reference code}', as illustrated in Fig. 7.

If the same 'reference code value' expression is intended to be used to identify other entities, then the identifier could have other types depending on the entity that it identifies. So, the type of the identifier on the document should be placed on the relation 'P_1 is identified by', as shown in Fig. 8.

But OWL2 only allows the use of binary properties, so this representation should be done as presented in Fig. 9 and follows the recommendation of CIDOC-CRM [31,33]: a subclass of 'PC_0 CRM Property' is created with the

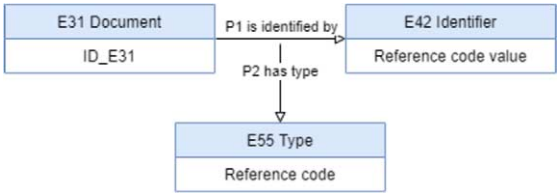


Fig. 8. Reference code representation with the type on the P_1 relation.

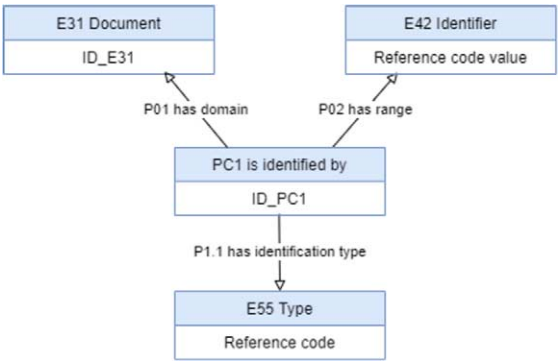


Fig. 9. Reference code representation with the type on the P_1 relation as binary relation.

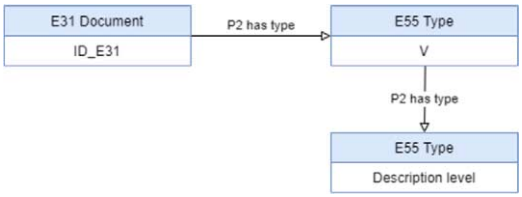


Fig. 10. Rule No. 2 description level.

name of the property that has a type, PC_1, 'PC_1 is identified by' with a new data property '$P_{1.1}$ has type'. The properties P_{01} and P_{02} are already defined in CIDOC-CRM, where class 'PC_0 CRM Property' is the domain and 'E_1 CRM Entity' is the range.

However and as explained before, the expression of 'reference code value' is a unique value and with the intention of allowing to search and retrieve information using the term 'reference code', the mapping description rule used for this element needs only to establish the type over the identifier, as illustrated in Fig. 7, and captured by the rule No. 3, shown in Table 5 (the Appendix). The representation illustrated in Fig. 9 is used when the identifier instance of 'E_{42} Identifier' can identify more then one instance and can have more then one type. For instance, 'PT' can be an instance of 'E_{41} Appellation' with type 'Country abbreviation' when it identifies a Place, or it can be an 'E_{41} Appellation' with type 'Person name abbreviation' when it identifies a Person. Whenever the same instance of the class 'E41 Appellation' identifies different entities, the type classification must be on the relation as it happens in rules No. 6, No. 7, No. 8, No. 13, No. 14 and No. 15, of Table 5 (the Appendix).

Consider now the element 'Description level' of an unit. Its value establishes the type of the unit, according to ISAD(G) model of the constituents description units of an archive (Fig. 1), such as Fonds, Sub-Fonds, Series, Sub-Series, File, Item, etc. As a result, it is considered that the 'Description level' is the only type property of the 'E_{31} Document' that represents the unit. So an instance of 'E_{55} Type' is created with the value of the unit type and this instance will have the type 'Description level', if those instances do not exist already. Additionally, the property 'P_2 has type' is added to link the 'E_{31} Document' to its type 'E_{55} Type' (see Fig. 10). This representation is adequate for the interpretation of natural language queries, since it makes explicit that the value is a 'Description level'. Note

that if an unit could have more then one type, then the type of the type should be the property '$P_{2.1}$ has type' (similar to the one explained before and showed in Fig. 9). The Mapping Description Rule No. 2 (Table 5, the Appendix) captures the translation of the 'Reference code' element into CIDOC-CRM representation.

According to the proposed representation of the unit of description, the ISAD(G) elements are linked to the classes that represent the unit in CIDOC-CRM, as follows:

- the 'E_{31} Document' is linked to the elements, such as reference code (Rule No. 3), description level (Rule No. 2), scope and content (Rule No. 6), recipient (Rule No. 7), title (Rule No. 8), date of creation (Rule No. 9), date of modification (Rule No. 10), hierarchy (Rule No. 17);
- the 'E_{22} Human-made Object' is linked to the elements, such as original numbering (Rule No. 5), date range (Rule No. 11), dimension and support (Rule No. 12), current keeper (Rule No. 13), country (Rule No. 14), producer (Rule No. 15), producer type (Rule No. 16);
- 'E_{33} Linguistic Object' is linked to the elements, such as language of materials (Rule No. 4).

The Mapping Description Rules are presented in Table 5, in the Appendix, including the formalism interpretation used, and they are also displayed in a diagram format for better understanding.

4. Automatic migration of archives metadata to OWL2

The DigitArq platform, as mentioned before, is supported by a centralized repository (named DigitArq database, from now on), which allows to store all the collected material in a well-structured organization determined by the archival representation. The automatic migration of DigitArq records into CIDOC-CRM[5] is then based on simple translation rules for the elements where there is a mapping between ISAD(G) and CIDOC-CRM. Some of these Mapping Description Rules were already introduced in the previous Section 3 and the Table 5 (the Appendix) presents a summary of the rules established for the migration process. However, there are elements of the ISAD(G), such as 'scope and content', that provide a semi-structured text with additional information to the ones established by the translations of the elements themselves. For these texts there is no direct mapping rules between concepts and the representation must be made by building new mapping rules, according to their structure.

The complete migration process is done in three main steps: 1) DigitArq Metadata Extraction; 2) Migration Process; and 3) Ontology Knowledge Discovery. At first step, the metadata to be represented in CIDOC-CRM are extracted from the DigitArq database. The second step represents the effective mapping process between the ISAD(G) elements and the CIDOC-CRM representations, and is made using the introduced Mapping Representation Rules. Finally, the third step refers to the interpretation of some pieces of text provided by some ISAD(G) elements and that are not yet represented in the CIDOC-CRM Ontology. This last step is done entirely over the information

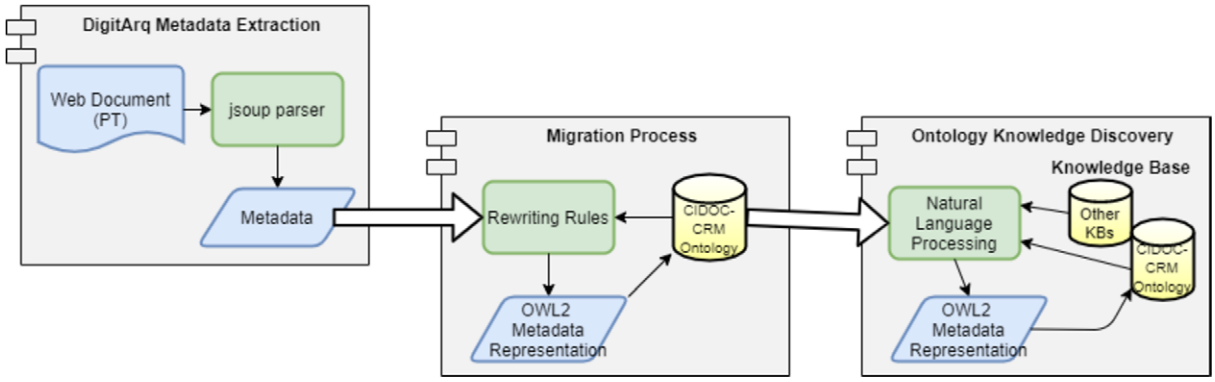


Fig. 11. Architecture for automatic migration of ISAD(G) units into CIDOC-CRM.

[5]CIDOC-CRM version 7 and its RDF Schema expression.

already represented in CIDOC-CRM, and obtained in the second step. The objective of the third step is to map valuable information to the knowledge base, by applying Natural Language Processing techniques to extract the additional information. Figure 11 presents the architecture of the migration process from DigitArq HTML records into CIDOC-CRM, the main tasks of each module are explained in the following subsections.

4.1. DigitArq metadata extraction

The DigitArq database contains a large and diverse amount of records, currently over 2 millions. As mentioned before, this database is structured, using a well-established standard archival description, with a hierarchical structure adapted to the nature of archival assets.

Along with the development of the DigitArq database, a web-based search engine (web service) was developed to allow local and remote users to find and browse the Archive's collections. The result is a well-structured and normalised web service[6] that for each unit shows the whole information needed to be considered in the migration process.

For this purpose, the `jsoup` library[7] is used to extract web page content from specific fields. `jsoup` is a Java HTML Parser that provides a very reliable, user-friendly, and easy configuration and parameter adjustments capabilities, for connecting to URLs and extracting and manipulating data.

The use of `jsoup` library to extract information from web pages to be analyzed and interpreted is not new, and can be found in [4,9]. The first one presents a solution for querying Greek governmental site, and the second one presents a solution to extract semi-structured information from web pages in the context of the innovation environments of the state of São Paulo, Brazil.

Each record's web page has a standardized scheme following the ISAD(G) and ISAAR(CPF) definitions, with the information organized according to a set of known fields and their values. Among this set of fields, there are some that present atomic values, such as "Reference code", "Title", or "Recipient", and others, that do not need further interpretation and the migration process is directly performed by applying the already introduced Mapping Description Rules (summarized in Table 5, the Appendix).

As an illustration, consider the fonds record, named "PARÓQUIA DE ALDOAR",[8] which describes the set of archival documents that composes it, regardless of its form or support, and concerning to baptisms, weddings, deaths, usages and customs, legacies and obligations of religious masses and indices registered by the Parish of Aldoar, from Oporto district, Portugal.

Using a set of `jsoup` functions, it is possible to extract information like the title, and other ISAD(G) elements (fields), as well as their values, presented in the web page record. For instance, the function *title()* allows to extract the title of the record; the function *getElementsByClass()* allows to extract the information per elements, as well as to get the set of its child records; the function *f.select("span").first().html()* allows to extract the name of each field and their values could be obtained using the function *text()*. `jsoup` also provides functions to connect and parse directly the web page source, such as *connect()* followed by *get()*, and *parse()*, respectively. The fragment of Java code Listing 1 illustrates how this is performed. Using such strategy to extract information, it is possible to consider other public webpage platforms as database that can offer the same or additional information, such as the "Archives Portal Europe".[9]

Some of the fields and the corresponding values extracted from "PARÓQUIA DE ALDOAR"'s fonds record are presented in Table 1.

The information extracted is adequately analysed, where each fields' name and their values are identified, the adequate ontology representation is established, and the corresponding ontology entities, such as individuals and properties, are then generated. This process is made by applying the set of Mapping Description Rules, some of them presented in Table 5(the Appendix) and introduced in the previous Section 3. Its implementation is explained with more detail in the following subsections.

[6] https://digitarq.arquivos.pt/

[7] http://jsoup.org

[8] http://pesquisa.adporto.arquivos.pt/details?id=488455, with dataset [23].

[9] The Archives Portal Europe provides access to information on archival material from different European countries as well as information on archival institutions throughout the continent. https://www.archivesportaleurope.net/home.

```
1 Document record = Jsoup.connect("http://pesquisa.adporto.arquivos.pt/details?id=488455").get();
2 String title = record.title());
3 Elements fields = record.getElementsByClass("Field");
4 for(Element f: fields) {
5   String field = f.select("span").first().html();
6   String value = f.text().substring(field.length()+1);
7 }
8 (...)
9 Elements childs = record.getElementsByClass("AspNet-TreeView-Leaf␣AspNet-TreeView-ParentSelected");
```

Listing 1. Metadata extraction Java code for the fonds record "PARÓQUIA DE ALDOAR".

Table 1

Example of some fields and the corresponding values extracted from "PARÓQUIA DE ALDOAR"'s fonds unit

Fields	Values
Description level	Fonds
Title	PARÓQUIA DE ALDOAR
Reference code	PT/ADPRT/PRQ/PPRT01
Title type	Formal
Date Range	1640-05-15 to 1911-03-31
Scope and content	Documentação relativa a baptismos, (...)
Creation date	05/22/2012 00:00:00

4.2. Migration process from DigitArq into CIDOC-CRM

The migration process consists of generating CIDOC-CRM ontology instances (classes and properties instances) by applying the Mapping Description Rules (see Section 3.3) to the DigitArq information.

At this step, the `OWL API`[10] [11] and the `SPARQL-DL`[11] [13,30] libraries are used to upload and model the CIDOC-CRM[12] [5,31] archival representation into a well-structured model for Java environment, to implement the mapping description rules, to update the mapping knowledge base, and also to reasoning over the knowledge base. The `OWL API` is a high level Application Programming Interface (API) for working with OWL ontologies, and is closely aligned with the OWL 2 structural specification.[13] It supports parsing and rendering in the syntaxes defined in the W3C specification, manipulation of ontological structures, and the use of reasoning engines. The `SPARQL-DL` is a Java query engine, settled on top of the `OWL API`, and it is fully aligned with the OWL2 standard and adds a `SPARQL-DL` interface to every `OWL API 3` reasoner.

Using the mentioned tools, the set of commands representing each mapping description rule is directly translated to Java instructions, which allows for automatically generate the CIDOC-CRM representation for each DigitArq record, and save it in OWL2 format.

As mentioned in the previous Section 4.1, each DigitArq database record, interpreted as a unit of description, has a well-known structure represented by a set of fields and their values, as well as their hierarchical relationship with other units, according to archival standards. As presented before, the migration process defines for each unit the application of:

1. Rule 1, Table 5 (the Appendix) – The unit itself is mapped into an instance of 'E_{31} Document' that documents (property 'P_{70} documents') a physical object (instance of 'E_{22} Human-Made Object') and refers to (property 'P_{67} refers to') a linguistic object (an instance of 'E_{33} Linguistic Object'). The corresponding set of commands introduced in (1), the Appendix, generates the set of Java instructions as shown in Listing 2.
2. Rule 17, Table 5 (the Appendix) – If the unit is composed by a collection of other units then using the property 'P_{106} is composed of' allows to represent the hierarchical relationship between the unit and the units that are

[10] http://owlapi.sourceforge.net/

[11] https://www.derivo.de/en/resources/sparql-dl-api/

[12] CIDOC-CRM version 7 and its RDF Schema expression.

[13] https://www.w3.org/TR/owl2-syntax/

```
String erlangen_crm = "http://erlangen-crm.org/200717/";
(...)
OWLIndividual IDE22 = newInst(erlangen_crm + "E22_Human-Made_Object");
add_prop(IDE31,erlangen_crm + "P70_documents",IDE22);
OWLIndividual instanceE33 = newInst(erlangen_crm + "E33_Linguistic_Object");
add_prop(IDE31,erlangen_crm + "P67_refers_to", IDE33);
```

Listing 2. "Java code translation of Rule No. 1".

```
String erlangen_crm = "http://erlangen-crm.org/200717/";
(...)
case "Description level":
  OWLIndividual ID_E55_1 = inst(erlangen_crm + "E55_Type", V);
  OWLIndividual ID_E55_2 = inst(erlangen_crm + "E55_Type", "Description level");
  prop(ID_E55_1, erlangen_crm + "P2_has_type", ID_E55_2);
  add_prop(instanceE31, erlangen_crm + "P2_has_type",ID_E55_1);
  break;
```

Listing 3. "Java code translation of Rule No. 2".

part of it. The hierarchical link between two units 'ID_{E31_1}' and 'ID_{E31_2}', instances of the 'E_{31} Document', is established by applying the following command

Prop(ID_{E31_1}, 'E_{31} Document', 'P_{106} is composed of', ID_{E31_2}, 'E_{31} Document')

with the following corresponding Java instruction

```
Prop(IDE31_1,"http://erlangen-crm.org/200717/P106_is_composed_of",IDE31_2);
```

The unit ID_{E31_2} is interpreted as a part of the unit ID_{E31_1} and with a description level below in the hierarchical tree representation (Fig. 1).

3. The remain rules, Table 5 (the Appendix) – For each ISAD(G) element that are described in the unit, the corresponding rule is applied to map the information into CIDOC-CRM representation. For instance, considering rule No. 2, which maps the 'Description level' of the unit, the corresponding set of commands, presented in (2), the Appendix, is translated to the Java instructions illustrated in Listing 3. The set of Mapping Description Rules applied varies according to the information that is described in the unit.

For better understanding, let us go back to the fonds unit "PARÓQUIA DE ALDOAR" and consider the elements 'Description level', 'Reference code', 'Language of the material', and 'Date range'. Figure 12 shows the solution obtained in the mapping process by applying respectively the rules No. 2, No. 3, No. 4 and No. 11 for the corresponding elements and their values. The mapping representation of the unit is obtained by applying the rule No. 1, and the mapping representation of the hierarchical relationship with other units is obtained by applying the rule No. 17. The fonds unit is composed by 8 other units (Fig. 12 just presents 2 of them), each one with the classification of 'Series' as 'Description level' (and each one is composed by other units [23]). The size of the hierarchy depends on the type and composition of the fonds and on what is described in DigitArq database. The complete representation of the fonds unit "PARÓQUIA DE ALDOAR" in CIDOC-CRM representation is obtained by applying the mapping process for each unit, belonging to the hierarchical tree of the fonds. Table 2 shows the mapping process metrics of the CIDOC-CRM representation of the fonds unit "PARÓQUIA DE ALDOAR", as well as its complete hierarchical composed units.

4.3. Ontology knowledge discovery

The Ontology Knowledge Discovery step consists of, by applying Natural Languages Processing (NLP) techniques, finding the proper interpretation of some text fields, from instances of CIDOC-CRM class, such as 'E_{62} String', that contain strings with description information as their value, and therefore extract additional information. The 'Scope and content' is one of the ISAD(G) elements that is characterized by having additional information describing its unit, and it is in text format.

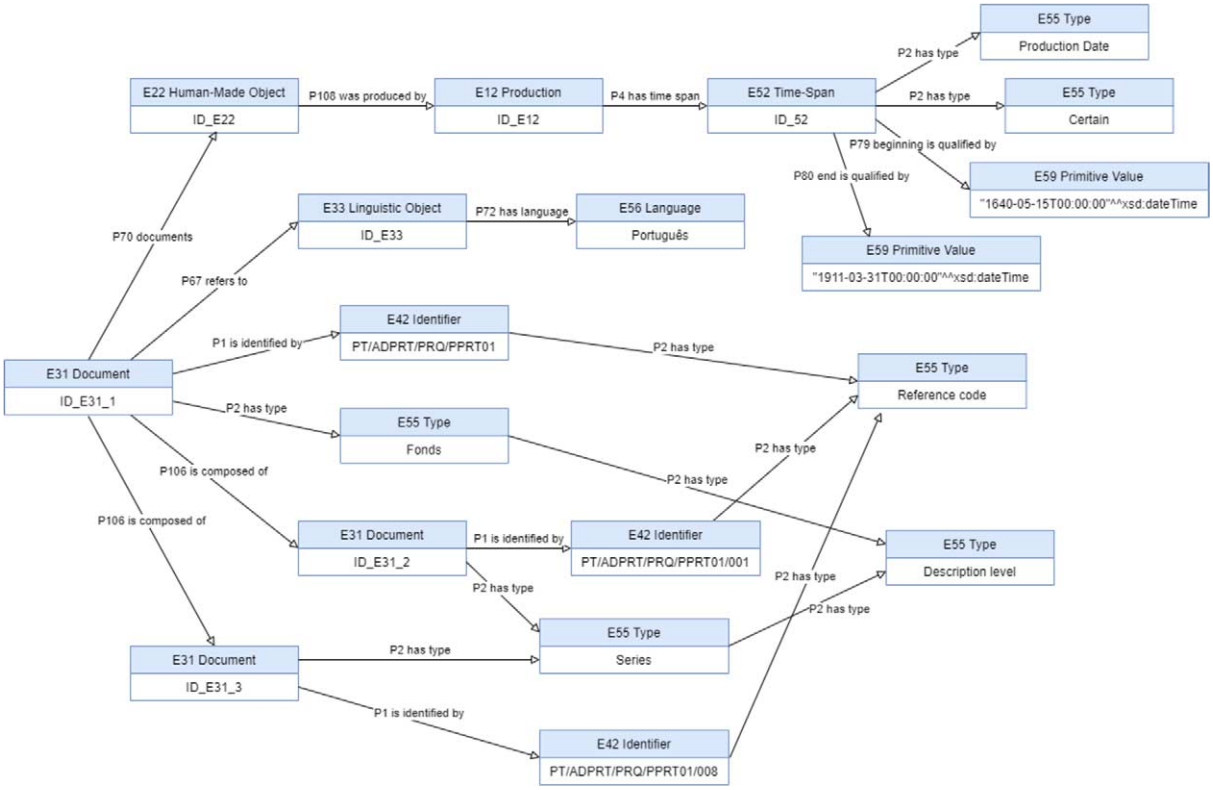


Fig. 12. The fonds unit "PARÓQUIA DE ALDOAR" partial mapping.

Table 2
Mapping process metrics of the complete CIDOC-CRM representation of the fonds "PARÓQUIA DE ALDOAR"

	Total
Axiom	146609
Logical axiom	106766
Declaration axioms	34058
Class	84
Object property	298
Data property	10
Annotation Property	4
Individuals	33666
Object property assertion between Individuals	72016

These texts, usually, have a structure that can be recognized, by using NLP tools, and giving as output a feature value list that will be the input of the migration sub-process. Enumeration is a structured pattern that is frequent in these text fields.

Consider as an illustration the fonds unit entitled "JUÍZO DA ÍNDIA E MINA",[14] which describes the set of archival documents that compose it, regardless of its form or support, and concerning civil and criminal processes registered under Portuguese discoveries root "Índia" and "Mina". Most of those processes are related to shipping damage, payment of soldiers, collection of freight, freight and unloading, qualification of heirs, processes of in-

[14] http://digitarq.arquivos.pt/details?id=4208377, with dataset [23].

Table 3
Populate metrics of the scope and content field of the "JUÍZO DA ÍNDIA E MINA"'s fond unit

Individual axioms	Scope and content	Migration step	Total
Class assertion	245	43	288
Object property assertion	381	44	425

dividuals who wanted to prove to be Portuguese and that their ships had made in Portuguese shipyards, with no foreigners interested in their cargo, and also about Corsair and piracy lawsuits. This fonds unit has the following piece of text withdrawn from its 'Scope and content' element.

> "Referem ainda o tipo de embarcações: navio, corveta, bergantim, galera, escuna, brigue, iate, caíque, nau, sumaca, barco, corsário, polaca"
>
> (They also mention the type of vessels: ship, corvette, brigantine, galley, schooner, brig, yacht, caique, ship, sumaca, boat, corsair, polish)

From this text using some NLP tools (e.g., tagger and lemmatization) and some grammar rules, it is obtained a list of Type-Value (vessel, name) elements. The representation of this information in CIDOC-CRM is given by rule No. 19, Table 6, the Appendix.

Each pair vessel-value gives rise to a new instance of 'E_{22} Human-Made Object' connected to the 'E_{31} Document' of the fonds unit by the Object property 'P_{129} is about', the instance 'E_{22} Human-Made Object' has type value that has type 'vessel'. Therefore, rules No. 19 to No. 21, from Table 6 (the Appendix), can be automatically generated from the text structure in ISAD(G) properties to obtain the CIDOC-CRM representation of the corresponding information.

For this example, Table 3 presents the total of axioms generated to represent the information interpreted, which increases substantially the amount of entities in the knowledge base compared to the entities generated in the migration process of the fonds unit itself. More important, it is possible to retrieve such information and infer about it, both automatically.

Another structure pattern that is often found in 'Scope and content' element is the identification of people and their relationship role with the 'Recipient' when referring to activities [20]. These activities can be baptisms, weddings or deaths, and the information is organized in a list of names and tagged by the role of the relationship that connects to the 'Recipient' of the unit. For instance, to illustrate this pattern, consider the Item unit entitled "REGISTO DE BAPTISMO",[15] which refers to the baptism happening of the person named "Ana", 'Recipient' of the unit, and its 'Scope and content' element has the following text value:

> "Pais: Manuel de Oliveira e Rufina Maria
> Avos maternos: Manuel da Fonseca e Rosa da Silva
> Avós paternos: José de Oliveira e Jacinta de Oliveira
> Padrinhos: Manuel Martins Ramos e Maria Francisca
> Data de nascimento: 10 de Fevereiro de 1812"
>
> ("Parents: Manuel de Oliveira and Rufina Maria
> Maternal grandparents: Manuel da Fonseca and Rosa da Silva
> Paternal grandparents: José de Oliveira and Jacinta de Oliveira
> Godparents: Manuel Martins Ramos and Maria Francisca
> Birthdate: 10th February, 1812")

Applying to this text some NLP tools (e.g. tagger and lemmatization) and some grammar rules, it is possible to extract the names of each person and the corresponding role of the relationship that links each person to "Ana" through the baptism activity. In particular, this means that it is possible to identify "Ana"'s parents, grandparents from both sides and also her birthdate.

[15] http://pesquisa.adporto.arquivos.pt/details?id=1374655

Unlike what happens with the birth event for which the CIDOC-CRM has the entity 'E_{67} Birth' to represent this concept, CIDOC-CRM model does not have a specific entity to represent the baptism activity. In this case, any baptism activity is established as an instance of the entity 'E_7 Activity' and to mark the activity as a baptism, it is applied a type to the activity, i.e., 'E_7 Activity' 'P_5 has type' 'E_{55} Type'.

To establish parenting relationship through the birth event, CIDOC-CRM model has proper properties, like 'P_{96} by mother', 'P_{97} by father' and 'P_{98} brought into life', that allow to perfectly represent the roles of being a father or a mother of someone else. To set a name to a person, the CIDOC-CRM has the class 'E_{41} Appellation' whose instances values represent names and the object property 'P_1 is identified by' to link the name to the person. For instance, to set 'Manuel de Oliveira' as the name of a person, the representation expression is 'E_{21} Person' 'P_1 is identified by' 'E_{41}{Manuel de Oliveira}'. The rule No. 22, in Table 6 (the Appendix), expresses the complete mapping description for a birth event, described in a baptism record unit.

The role of being a godparent is established through the baptism activity and the CIDOC-CRM model does not have a proper set of entities and properties that explicitly represent those concepts. The solution is to establish a ternary relation where the role of the relationship is expressed as the type of a property. The entity 'PC_{14} Carried Out By' represents the relationship, which has type ('$P_{14.1}$ has type') the role of the relationship ('E_{55} Type'), with domain the baptism 'E_7 Activity' and range the godparent 'E_{21} Person'. Rule 23, in Table 6 (the Appendix), captures the complete mapping description for the godparents relationship.

Figure 13 shows the complete solution for the baptism example, obtained after the application of the Mapping Description Rules identified and expresses the axioms that are added to the knowledge base.

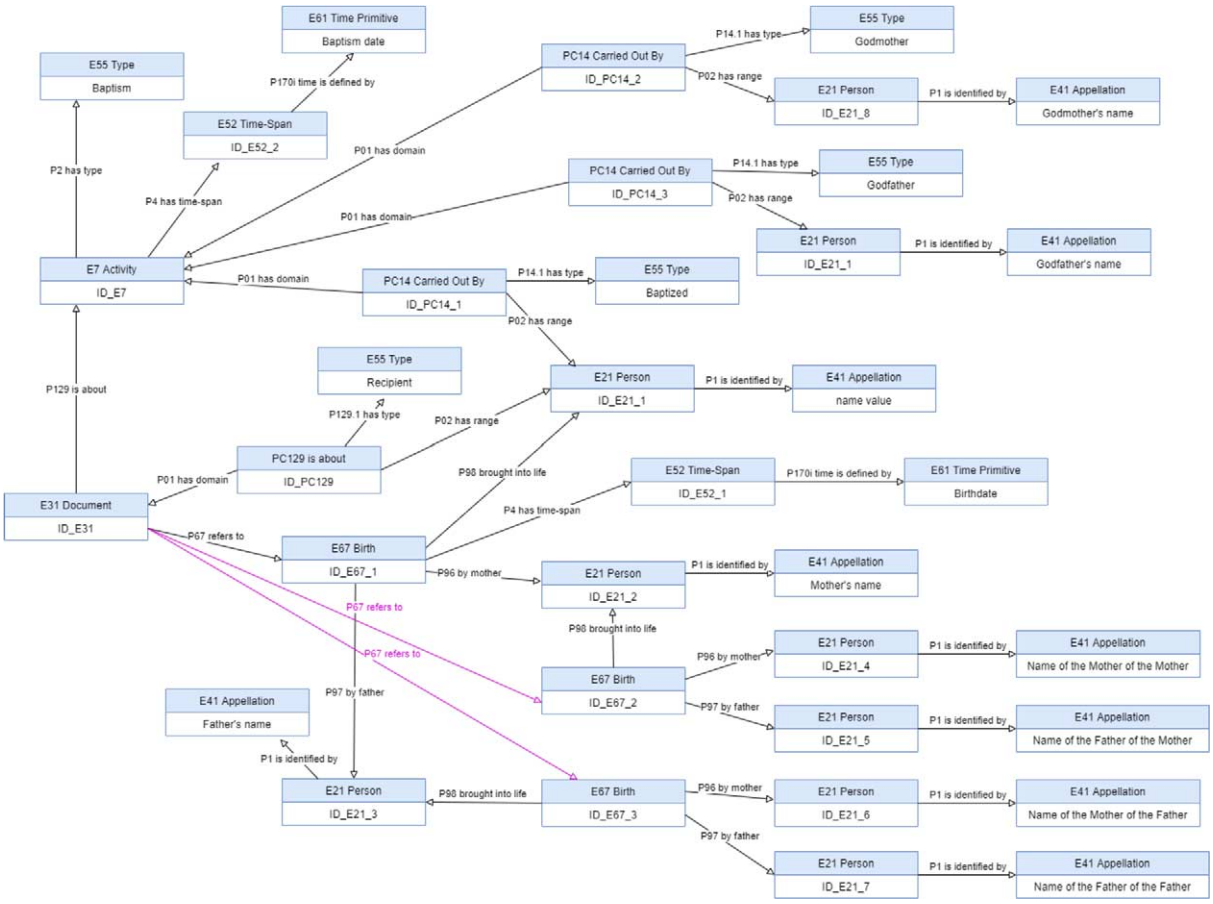


Fig. 13. The baptism activity of the person "Ana" partial mapping.

In the Appendix, the Mapping Description Rules, presented in Table 6, are also displayed in a diagram format for better understanding.

4.4. Evaluation of migration process

The Migration Process is composed of 3 modules (Fig. 11), as presented in previous subsections. The development state of each module is the following:

The first module refers to the information extraction from DigitArq webpages, using `jsoup` tool. This task is complete, which means that each ISAD(G) element presented in a webpage is extracted in the form as presented in Table 5 (the Appendix), second column. Therefore, the information extracted matches the original one.

The evaluation of this process was made at the same time as the development, where the tests were made manually by comparing the input webpage with the corresponding output. The evaluation results grants that the process has a 100% accuracy, which means that the information extraction from DigitArq is correct.

The second module consists of the migration process of the previous output information to the CIDOC-CRM representation, according to the rules presented in Table 5 (the Appendix). This task is complete, which means that all the rules are implemented and it is possible to represent all the units of description in CIDOC-CRM.

The evaluation of this process is made automatically. For each unit of description represented in DigitArq, it is possible to compare the output obtained at first module with the information recovered in CIDOC-CRM Knowledge Base. This process is made by using `SPARQL-DL` queries to search in the OWL Knowledge Base. For instance, to check that the unit with reference code "PT/ADPRT/PRQ/PPRT01/001/0004/00001" has as description level value 'Item', it is only necessary to question the knowledge base with the DL query:

DLq = *inverse* 'P_2 has type' *some* ('P_1 is identified by' *value*
'PT/ADPRT/PRQ/PPRT01/001/0004/00001')

Answer to DLq: 'Item'

The answer to this DL query, 'Item', is compared to the 'Description level' value obtained in the first module for the same unit of description, which is ['Description level', 'Item'].

The evaluation process was made for all the elements of ISAD(G) presented in DigitArq web interface for a sample of fonds, and their hierarchical composition, for instance, the "Paróquia de Aldoar" with a total of 1426 units of description. The evaluation results grants that the process has a 100% accuracy, which means that the implementation is correct.

The third module consists of representing the information contained in textual elements of ISAD(G), such as 'Scope and content'. This task has as input the instances of the Knowledge Base that are Strings, representing texts written in Portuguese, from where some information must be extracted, such as baptisms, births, incorporations of material into Fonds, transfers of materials, lists of entities, father, mother, godparents, etc.

This process is defined in 3 sub-process: text classification, extraction, and representation. All these sub-processes are ongoing and no formal evaluation has been made yet.

Regarding the representation sub-process, some of the information to extract has already been identified. For this information, a set of rules (Table 6, in the Appendix) is already defined and evaluated with a sample. This evaluation was made automatically, by using `SPARQL-DL` queries to search in the OWL Knowledge Base, and explained in the next Section 5. This task is not complete, since it is necessary to identify more information to be extracted and to define the rules to represent it in CIDOC-CRM.

In the text classification sub-process, it is defined a classifier for each type of information to identify the texts where that kind of information could be extracted. These classifiers are built using machine learning and natural languages processing tools. For instance, to identify the texts that contain baptism information, it was built a manually annotated sample of 200 Portuguese texts and a classifier using a `N-Gram TF-IDF` model for the sample data and a decision tree, allowing to obtain a high precision in the identification of the texts containing information about baptisms.

In general, the results, obtained with the classifiers, were not so good because the recall had lower values, but the precision had high values. However, these results are considered at this moment good enough to classify the text. As future work, it is intended to use other language models to improve the recall of the classifier.

About the extraction sub-process, in the classified text, the information is extracted into an established format (column 2, Table 6, in the Appendix). A pipeline of natural language processing tools was integrated in the Gate framework.[16] A set of Jape rules was also used to extract the entities and relations. This work is ongoing but some experiences were already performed, enabling the automatic extraction of some information, and can be confirmed in the dataset example [24].

4.5. Migration to other ontologies

In this subsection, a set of rules to map ISAD(G) into RIC-O, following the proposed strategy representation, is presented to show that the migration process can be easily adapted to the use of other ontologies.

The migration of DigitArq into RIC-O could be achieved by writing the new mapping description rules that model the Portuguese National Archives in RIC-O. As an example, it is presented in Figs 14, 15, 16, and 17, and Table 7 in the Appendix, the corresponding rules in RIC-O for the elements 'Unit of Description', 'Description level', 'Reference code', 'Language of the material' and 'Hierarchy'. For instance and as can be seen in Rule No. 2 (Fig. 10 in Section 3.3), the 'Description level' representation in CIDOC-CRM is more expressive than the one in RIC-O. CIDOC-CRM uses general concepts to describe the elements and it is possible to assign specific types to those elements. In RIC-O model, classes and properties are more specific and related to the ISAD(G) elements, such as 'rico:RecordSetType' or 'rico:hasRecordSetType', and there is no property that allows to apply a type (category) to a 'rico:RecordSetType'.

The representation of the unit of description in RIC-O (see Fig. 14) is set with an instance of the class 'rico:RecordSet' and linked with a new instance of the class 'rico:Instantiation' through the property 'rico:hasInstantiation'.

Similar to Rule No. 2 in Fig. 15, to represent 'Reference code', RIC-O model uses specific classes and properties to represent identifiers of a unit of description, such as 'rico:Identifier' and 'rico:hasOrHadIdentifier', and there is no property that allows to apply a type (category) to a 'rico:Identifier', see Fig. 16.

Fig. 14. Rule No. 1: Unit of description representation in RIC-O.

Fig. 15. Rule No. 2: Description level in RIC-O.

Fig. 16. Rule No. 3: Reference code in RIC-O.

Fig. 17. Rule No. 4: Language of the material in RIC-O.

[16] https://gate.ac.uk/

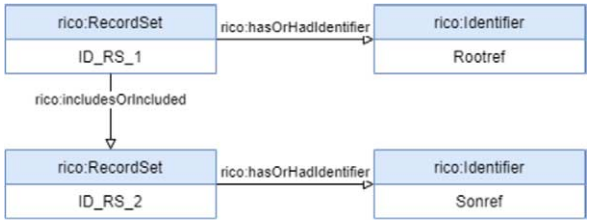


Fig. 18. Rule No. 17: Hierarchy in RIC-O.

For the case of the element 'Language of the material', the representation in RIC-O has the same expressiveness as CIDOC-CRM, see Fig. 17 and Fig. 22(a) in the Appendix.

To conclude, the hierarchical structure of the unit of description collection is modeled the same way as in CIDOC-CRM, see Fig. 18. Table 7, in the Appendix, shows CIDOC-CRM and RIC-O Mapping Description Rules, based on the figures previous mentioned, that are used for the automatic migration.

5. Querying CIDOC-CRM representation of archives metadata

The result of the migration process can be manually evaluated by querying the knowledge base, consisting of CIDOC-CRM Ontology and the complete set of assertions obtained through the Migration Process and the Ontology Knowledge Discovery.

The semantic web representation of the archival information will allow to explore the information in new ways, such as a chronological view of the archival materials, their geographical places, people related to the archival materials, etc. In the legacy system DigitArq, a relational model database, such exploration is difficult to obtain, even when using full string search.

As mentioned before, the Portuguese National Archives have their databases organized by Regional archives and are not integrated, which requires the search in each database. The semantic web model enables to integrate the information of all databases. It is also possible to integrate information from other archives, national or international, represented in CIDOC-CRM, by using the name of known entities to link data.

The information extracted from the text elements allows to represent new information, such as births, incorporations or transfers of documents, that were not presented in the relational model of the DigitArq database.

The guarantee that the CIDOC-CRM Ontology representation of the DigitArq metadata is well-performed is established, when questioning (searching) the knowledge base, it is possible to retrieve the original information.

5.1. *Querying the knowledge base*

The process of retrieving the information about the archival units uses the Mapping Description rules, presented in Table 5 and Table 6, in the Appendix, to define the Description Logic (DL) queries on the subject of a question.

The examples, presented in Table 4, illustrate queries to obtain some of the elements of an unit,[17] such as Reference code, Description level, Title, Title Type, Creator (Producer), Date Range.

To help the evaluation of the CIDOC-CRM representation of the migrated data, a web interface was developed, see Section 5.2 for further information. The queries Q8 and Q9, presented in Table 4, are examples of smart queries enabled by the interface application and that allow to explore the advantages of the archives OWL representation. For instance, with question Q9, the following examples of answers are obtained when questioning the knowledge base about the type of a given name:

- Name=Ana
  'E_{55} Type': Recipient

[17] These queries were done in Protegé with the reasoner Pellet over the dataset [24].

Table 4
DL-queries examples

Q1:	The unit with 'Reference code' PT/ADPRT/PRQ/PPRT01/001/0004/00001 DLq_1 = `'P1 is identified by' some (inverse 'P1 is identified by' some ('P1 is identified by' value 'PT/ADPRT/PRQ/PPRT01/001/0004/00001') and 'P2 has type' value 'Reference code')` Answer: E31_Document20
Q2:	'Description level' of the unit with 'Reference code' PT/ADPRT/PRQ/PPRT01/001/0004/00001 DLq_2 = `inverse '`P_2 `has type' some` DLq_1 `and '`P_2 `has type' value Description level` Answer: Item
Q3:	'Title' of the unit with 'Reference code' PT/ADPRT/PRQ/PPRT01/001/0004/00001 DLq_3 = `inverse '`P_{02} `has range' some ('`PC_{102} `has title' and '`P_{01} `has domain' some` DLq_1) Answer: Registo de Baptismo
Q4:	'Title type' of the unit with 'Reference code' PT/ADPRT/PRQ/PPRT01/001/0004/00001 DLq_4 = `inverse '`$P_{102.1}$ `has type' some ('`PC_{102} `has title' and '`P_{01} `has domain' some` DLq_1) Answer: Formal
Q5:	'Creator (the producer)' of the unit with 'Reference code' PT/ADPRT/PRQ/PPRT01/001/0004/00001 DLq_5 = `inverse '`P_{02} `has range' some (inverse` P_{01i} `is domain of' some (inverse '`P_{14} `carried out by' some ('`P_{108} `has produced' some (`P_{70i} `is documented in' some` DLq_1)))) Answer: PPRT01 *and* Paróquia de Aldoar
Q6:	'Date range' of the unit with 'Reference code' PT/ADPRT/PRQ/PPRT01/001/0004/00001 DLq_6 = `inverse '`P_{79} `beginning is qualified by' some (inverse '`P_4 `has time-span' some ('`P_{108} `has produced' some ('`P_{70i} `is documented in' some` DLq_1))) Answer: "1811-07-07T00:00:00"^^xsd:dateTime
Q7:	'Reference code' of units that mention 'corveta' DLq_7 = `inverse '`P_1 `is identified by' some ('P129 is about' some ('E22 Man-Made Object' and 'P2 has type' value 'corveta'))` Answer: `PT/TT/JIM`
Q8:	Instances identified by **Name** DLq_8 = `('`PC_1 `is identified by' and '`P_{02} `has range' value` **`Name`**`) or ('`P_1 `is identified by' value` **`Name`**`)`
Q9:	What is the type of **Name** DLq_9 = `inverse '`P_2 `has type' some (inverse` P_{01} `has domain' some ( '`PC_1 `is identified by' and '`P_{02} `has range' value` **`Name`**`)) or (inverse '`P_2 `has type' value` **`Name`**`) or (inverse '`P_2 `has type' some ('`P_1 `is identified by' value` **`Name`**`)) or (inverse '`$P_{0.1}$ `has type' some ('`P_{02} `has range' some ('`P_1 `is identified by' value` **`Name`**`))) or (inverse '`$P_{0.1}$ `has type' some (`P_{02} `has range' value` **`Name`**`))`

- Name=Paróquia de Aldoar
  'E_{55} Type': Institution name
- Name=Portugal
  'E_{55} Type': Country name
- Name=PT
  'E_{55} Type': Country abbreviation
  'E_{55} Type': Institution abbreviation[18]

Note that Protegé does not allow to query for a property of an instance, but it is possible to do it with `SPARQL-DL` as in Listing 4. This query retrieves all the instances of 'E_{31} Document' where its 'Description level' is 'Fonds' (Mapping Description Rule No. 2, Table 5, in the Appendix).

[18] PT is the Contry abbeviation of 'Portugal' and also the Institution abbreviation 'Portugal Telecom'.

```
PREFIX erlangen-crm: <http://erlangen-crm.org/200717/>
SELECT ?x WHERE {
  Type(?x,erlangen-crm:E31_Document),
  PropertyValue(?x,erlangen-crm:P2_has_type,?instance),
  Type(?instance,erlangen-crm:E55_Type>),
  PropertyValue(?instance,erlangen-crm:P2_has_type,erlangen-crm:Description_level),
  SameAs(?instance,erlangen-crm:Fonds)
}
```

Listing 4. "Units with 'Description level' 'Fonds'".

```
PREFIX erlangen-crm: <http://erlangen-crm.org/200717/>
SELECT DISTINCT ?p WHERE {
  Type(?x,erlangen-crm:E31_Document),
  PropertyValue(?y,erlangen-crm:P2_has_type,erlangen-crm:Reference_code>),
  ObjectProperty(?p),
  PropertyValue(?x,?p,?y)
}
```

Listing 5. "Properties that link an 'E_{31} Document' to a 'Reference code'".

These 'smart' queries are useful in the interface application, not only for helping the users to explore the knowledge base, but also to use in the interpretation of natural language text and assign ontology terms to sentences tokens.

Another kind of query that is important for the natural language interpretation process is to obtain object properties that link instances from a class domain and instances from a class range. The query presented in Listing 5 is an example of an SPARQL-DL query to obtain the object properties that link instances of 'E_{31} Document' to a 'Reference code'. This query can not be done in Protegé.

5.2. *Query ontology interface*

The knowledge base querying process is supported by an application program interface (API), entitled Query Ontology Interface, that facilitates the interaction between project team developers and the knowledge base. The main goals for the development of such API are to allow retrieving information from knowledge base without technically know how the information is represented in the ontology, as well as to express queries as near as possible to natural language text. The interface main target users, the EPISA project members that work in DGLAB, are in general not able to make queries using SPARQL language, or even using description logic languages.

The Query Ontology Interface has an important role in the process of manual evaluation during the development of the migration process, enabling the visualization of the extracted information, and helps in identifying the correctness of the information extracted and how it is related with the other ontology information.

The Query Ontology Interface was developed using Spring Boot,[19] a Java-based framework that allows to create a Graphical User Interface (GUI) and export the final API in a stand-alone application (originally a web-application). The SPARQL-DL Java query engine is used to search the knowledge base, and serves as a middle layer application between the GUI and the knowledge base. The question made by the user at the GUI level is translated to the corresponding CIDOC-CRM representation and the answer is retrieved using the SPARQL-DL engine and then presented at the GUI application level. The approach used to query the knowledge base is as much user-friendly as possible, besides the use of a GUI, it also uses Natural Language understanding mechanisms to help the users in the querying process. Other existing interfaces over CIDOC-CRM Ontology use similar approach, such as OpenArcheo [17].

The Query Ontology Interface is able to retrieve information about single individuals and about the structure of the whole knowledge base. For instance, it is possible to retrieve information based on the value of some key-entities, like 'E_{41} Appellation' or 'E_{42} Identifier', that are expressed in Natural Language and stored as xsd:string.

[19] https://spring.io/

(a) Query the ontology about a specific 'Reference code' value

(b) The query result for a specific 'Reference code' value.

Fig. 19. Query ontology interface example.

A user expression can be interpreted as a class or as a class constrained by a property. As an example, consider a search using the expression "Ana", the result is a DL-query where a person is constrained to be identified by an appellation with value "Ana". This NL interpretation tool is still in development.

It is also possible to define a constraint (or a joint of constraints) to retrieve the desired individuals. The result of such query is an individual (or a joint of individuals) with all the properties and other individuals linked to it. In addition, it was defined a set of predefined queries that work like filters, such as displaying all the class entities or all the individuals belonging to a class entity. An example on how it works is shown in Fig. 19, where the search is made using the 'Reference code' value, see Fig. 19(a), and the result is the information of the corresponding unit elements, see Fig. 19(b).

6. Open problems

The development of the Mapping Description Rules and their implementation process, together with the analysis of different examples, allowed to notice some issues that led to a set of open problems.

In the Ontology Knowledge Discovery process, two of the major problems identified are, first, to know exactly the information available in the text elements and, second, what is possible and important to infer from them. The text fields are free text, but depending on what they are about, what event or subject they are describing, it is possible to identify some structure which allows to define proper mapping description rules for their representation. For instance, consider the example of "Ana"'s baptism unit presented in Section 4.3 and the semi-structured text of its 'Scope and content' element. The text format happens to be equal for all the units referring to baptism activities. After knowing the subject type of the unit, the information available in the text fields can be represented by applying the mapping description rules established for the corresponding semi-structured text, as explained before. For the "Ana"'s baptism unit and its 'Scope and content' value example, the axioms generated by the Mapping Descriptions Rules from No. 22 to No. 26, from Table 6 (the Appendix), depends on the information available. For instance, if the information about the grandparents is not available, it is not possible to infer the representation of the birth events for both "Ana"'s parents, then the corresponding axioms are not added to the knowledge base. However, if the information to generate the axioms, even being complete, is not correctly interpreted and identified, it may lead to inaccurate representation. Therefore, proper NLP techniques are necessary to make the adequate interpretation and identification of the information available, allowing, beyond the application of the correct mapping description rules, to generate accurate information representation.

Looking in particular to this example, some other questions occurred, beyond the simple interpretation of the text. For instance, when a new person shares some properties, such as the name or different variant names, with a known person, should it be considered that it is the same person? when a person has the same names for its parents and grandparents of a known person in the knowledge base, are they siblings? should that relationship role be considered and added to the knowledge base?

At this point, it was decided to consider that when two persons are identified by the same name or different variant names, they are distinct persons. The same strategy was followed with other entities, such as places, times, and other instances with an identification. This enables to define a disambiguation process, a posteriori. This process could

be done: automatically by defining new axioms, such as 'countries with the same name are the same country', or 'two persons with the same name, and the same father and mother names, are the same person'; or manually by a specialist which asserts, for instance, that those two persons are the same person.

In the process of interpreting and representing enumeration lists, the following issues were identified and need to be taken into account:

- Synonyms – in enumerations the Type or the value can be a word or phrase that means exactly or nearly the same as another word or phrase that was already introduced as a new 'E55 Type´, e.g. vessel and ship.
- Names – in the same document or same enumeration list a name can appear more than once.

With regard to the Mapping Description Rules, from Table 5 (the Appendix), there are some exceptions, presented below, that need to be considered in the process of extracting the information necessary for the migration process.

The Mapping Description Rule No. 7 correspond to the 'Recipient' representation, main entity to which the unit refers to. The 'Recipient' element could not be explicitly presented in the description of the unit and, when it happens, the ISAD(G) element that could provide this information is the 'Title' element of the unit. For these cases, the 'Title' element should be properly interpreted by applying NLP rules that allow to identify the title and the recipient of the corresponding unit. For instance, the unit with reference code 'PT/ADPRT/PRQ/PPRT04/001/0054/000013' does not have a 'Recipient' element, but its title 'REGISTO DE BATISMO DE ANA' (ANA BAPTISM REGISTRATION) includes the recipient name 'Ana'.

The Mapping Description Rules No. 13 and No. 14 are applied to explicitly represent, respectively, the current keeper and the current location country of the physical object described in a unit. When this information is not explicitly represented with an adequate unit element, it is possible to extract the information required from the 'Reference code' value of the unit. As mentioned before, the 'Reference code' element identifies uniquely the unit of description. To provide an accurate link to the information of the unit, the following conditions are taken into account when creating the 'Reference code' value of an unit: first, the country code in accordance with the latest version of ISO 3166 Codes for the representation of countries names; second, the repository code in accordance with the national repository code standard or other unique location identifier; and third, a specific local reference code, control number, or other unique identifier.

The Mapping Description Rules No. 15 and No. 16 capture the representation of the 'Creator' of the unit and its type. When these information values are not explicitly available to be interpreted, the 'Reference code' value of the unit also provides the information required to be interpreted.

As an example, consider the reference code 'PT/ADPRT/PRQ/PPRT04/001/0054/000013', where:

PT is the country abbreviation of Portugal.
ADAVR is the keeper abbreviation name of Arquivo Distrital do Porto.
PRQ is the abbreviation of the institution type, Paróquia (parish), of the producer.
PPRT04 is the abbreviation of Paróquia de Cedofeita (Cedofeita parish), the producer or creator of the unit.

An adequate interpretation of the 'Reference code' value a unit will allow to extract and then represent the information about the current keeper and current location of the corresponding unit, as well as its creator and the type of the creator.

The Mapping Description Rule No. 18 expresses the relationship between the 'Original numbering' identification and the location country, i.e, the first place 'falls within' the second place. This interpretation only occurs when the unit explicitly presents the 'Original numbering' value and can only be set after the current location country of the physical object of the unit is already represented.

The Mapping Description Rule No. 14 associates a country to a human-made object. The object property 'P_{55} current location' has the constrain of max. 1, implying that there is at most one place. Therefore, to avoid the unification of different instances of place, whenever a country place is used, a new instance of 'E_{53} Place', the second in this rule, is created with the same linked properties apart the link to the first place. With this rule, the migration of a fond with hundreds of units will give rise to hundreds of 'E_{53} Place' to represent the (same) country. It is possible to avoid this proliferation of similar instances by correcting the Mapping Description Rule No. 14 or add a new Mapping Description Rule similar to rule No. 18 that will infer that those instances are the same.

The expressiveness power of the Mapping Description Rules with the proposed extensions is enough to deal with this kind of issues.

7. Conclusions and future work

The experience results show that the use of Mapping Description Rules with the proposed extensions has the expressiveness power necessary to define the representation of structured information, such as archives, in an OWL2 ontology such as CIDOC-CRM. These Mapping Description Rules can be automatically interpreted using an environment, such as `OWL API`, to obtain the set of assertions that represents the information in the target ontology.

The task of representing the information, such as an archive, in an ontology requires the study of the ontology and their recommendations in order to achieve interoperability sharing and to use information already represented in the ontology, as well as the use of platforms to explore the information represented. The use of CIDOC-CRM model is a guarantee that, on the one hand, there are already many information available in the area of cultural patrimony that can be used to integrate and linked with, and on the other hand, there are also many platforms available that can be used to explore the information migrated.

Another important issue when representing information in an ontology is to take into account the need of interpreting natural language text, to automatically obtain its ontology representation. Like in this subject domain, archives information, free text appears in a variety of metadata fields of other domains. Interpreting natural language text can condition the representations in the ontology as presented in regard to this work.

Some examples were presented about the migration of the metadata information within text fields, but currently this task is under development in order to achieve the automatic migration of events, persons, institutions, places, etc.

Regarding the migration process evaluation there are two sub processes, the set of mapping description rules presented in Table 5 and the set of rules from Table 6 (the Appendix). For the first one, the result migration either is correct or not, if the information retrieved from OWL2 representation is successfully matched with the initial records, then it is correct. Otherwise, it is necessary to identify the problems in mapping representation and then they should be fixed. This evaluation can be made automatically, but an application interface as the one presented is helpful to debug the problems that can occur. For the second set of rules, the evaluation is more complex and requires human intervention to decide if the information extracted from the text fields is well represented and relevant. This evaluation is not done yet and, at this moment, the interface application only retrieve information represented for each unit, obtained in the first step of the migration process. This task is set as future work.

Acknowledgements

This work is financed by National Funds through FCT – Foundation for Science and Technology I.P., within the scope of the EPISA project – DSAIPA/DS/0023/2018.

Appendix. Mapping description rules

The current Appendix is used to introduce the Mapping Description Rules and to display them in a diagram format for better understanding.

A.1. Formalism of mapping description rules

This Subsection presents the formalism of Mapping Description Rules used in the migration process of ISAD(G) elements into CIDOC-CRM representation. The Mapping Description Rules are summarized in Table 5 and Table 6.

As presented previously throughout the paper, the ISAD(G) elements of each unit of description can be grouped, according to their content and what they refer to, namely the object itself that belongs to the physical archive; the digital registration that describes the object; and the language properties associated to the object (when they exist). These three concepts are mapped into the CIDOC-CRM classes, respectively 'E_{22} Human-Made Object'; 'E_{31} Document'; and 'E_{33} Linguistic Object'. Whereby, each unit of description is itself mapped into an instance

Table 5
Mapping description language rules

Rule No.	Left part (rec[attribute value list])	Right part CIDOC-CRM
1	DigitArq(Rec)	$E_{31}\{=ID_{E_{31}}\} \rightarrow P_{70} \rightarrow E_{22}\{=ID_{E_{22}}\} \rightarrow P_{67} \rightarrow E_{33}\{=ID_{E_{33}}\}$
2	['Description level', V]	$\$ID_{E_{31}} \rightarrow P_2 \rightarrow (\langle E_{55}\{=V\}\rangle \rightarrow \langle P_2\rangle \rightarrow \langle E_{55}\{\text{= 'Description level'}\}\rangle)$
3	['Reference code', V]	$\$ID_{E_{31}} \rightarrow P_1 \rightarrow (E_{42}\{=V\} \rightarrow P_2 \rightarrow \langle E_{55}\{\text{= 'Reference code'}\}\rangle)$
4	['Language of the material', V]	$\$ID_{E_{33}} \rightarrow P_{72} \rightarrow \langle E_{56}\{=V\}\rangle$
5	['Original numbering', V]	$\$ID_{E_{22}} \rightarrow \langle P_{55}\rangle \rightarrow (\langle E_{53}\rangle \rightarrow P_{89} \rightarrow (E_{53} \rightarrow P_1 \rightarrow \langle E_{41}\{=V\}\rangle \rightarrow P_2 \rightarrow \langle E_{55}\{\text{= 'Original numbering'}\}\rangle))$
6	['Scope and content',V]	$\$ID_{E_{31}} \rightarrow P_{01i} \rightarrow (PC_3 \rightarrow P_{02} \rightarrow \langle E_{62}\{=V\}\rangle \rightarrow P_{3.1} \rightarrow \langle E_{55}\{\text{= 'Scope and content'}\}\rangle)$
7	['Recipient', V]'	$\$ID_{E_{31}} \rightarrow P_{01i} \rightarrow (PC_{129} \rightarrow P_{02} \rightarrow (E_{21} \rightarrow P_1 \rightarrow \langle E_{41}\{=V\}\rangle) \rightarrow P_{129.1} \rightarrow \langle E_{55}\{\text{= 'Recipient'}\}\rangle)$
8	['Title and Type',(T_i, T_y)]	$\$ID_{E_{31}} \rightarrow P_{01i} \rightarrow (PC_{102} \rightarrow P_{02} \rightarrow \langle E_{35}\{=T_i\}\rangle) \rightarrow P_{102.1} \rightarrow \langle E_{55}\{=T_y\}\rangle)$
9	['Creation date', V]	$\$ID_{E_{31}} \rightarrow \langle P_{94}\rangle \rightarrow (E_{65} \rightarrow P_4 \rightarrow (E_{52} \rightarrow P_2 \rightarrow \langle E_{55}\{\text{= 'Creation date'}\}\rangle \rightarrow P_{170i} \rightarrow \langle E_{61}\{=V\}\rangle))$
10	['Modification date',V]	$\$ID_{E_{31}} \rightarrow \langle P_{94}\rangle \rightarrow (E_{65} \rightarrow P_4 \rightarrow (E_{52} \rightarrow P_2 \rightarrow \langle E_{55}\{\text{= 'Modification date'}\}\rangle) \rightarrow P_{170i} \rightarrow \langle E_{61}\{=V\}\rangle))$
11	['Date Range', (I,F,C)]	$\$ID_{E_{22}} \rightarrow \langle P_{108i}\rangle \rightarrow (E_{12} \rightarrow P_4 \rightarrow (E_{52} \rightarrow P_2 \rightarrow \langle E_{55}\{=C\}\rangle \rightarrow P_{79} \rightarrow \langle E_{61}\{=I\}\rangle \rightarrow P_{80} \rightarrow \langle E_{61}\{=F\}\rangle \rightarrow P_2 \rightarrow \langle E_{55}\{\text{= 'Production Date'}\}\rangle))$
12	['Dimension and support', (V_s, D_u, V_d)]	$\$ID_{E_{22}} \rightarrow P_{45} \rightarrow \langle E_{57}\{=V_s\}\rangle \rightarrow P_{43} \rightarrow (E_{54} \rightarrow P_{91} \rightarrow \langle E_{58}\{=D_u\}\rangle \rightarrow P_{90} \rightarrow \langle E_{60}\{=V_d\}\rangle)$
13	['Current keeper', (V_{abv},V_n)]	$\$ID_{E_{22}} \rightarrow P_{50} \rightarrow (E_{74} \rightarrow P_{01i} \rightarrow (PC_1 \rightarrow P_{02} \rightarrow \langle E_{41}\{=V_{abv}\}\rangle \rightarrow P_{1.1} \rightarrow \langle E_{55}\{\{\text{= 'Institution abbreviation'}\}\}\rangle) P_{01i} \rightarrow (PC_1 \rightarrow P_{02} \rightarrow \langle E_{41}\{=V_n\}\rangle \rightarrow P_{1.1} \rightarrow \langle E_{55}\{\text{= 'Institution name'}\}\rangle))$
14	['Country', (V_{abv}, V_n)]	$\$ID_{E_{22}} \rightarrow P_{55} \rightarrow \langle P_{55}\rangle \rightarrow (\langle E_{53}\rangle \rightarrow P_{89} \rightarrow (E_{53} \rightarrow P_2 \rightarrow \langle E_{55}\{\text{= 'Country'}\}\rangle \rightarrow P_{01i} \rightarrow (PC_1 \rightarrow P_{02} \rightarrow \langle E_{41}\{=V_{abv}\}\rangle \rightarrow P_{1.1} \rightarrow \langle E_{55}\{\{\text{= 'Country abbreviation'}\}\}\rangle) P_{01i} \rightarrow (PC_1 \rightarrow P_{02} \rightarrow \langle E_{41}\{=V_n\}\rangle \rightarrow P_{1.1} \rightarrow \langle E_{55}\{\text{= 'Country name'}\}\rangle)))$
15	['Producer', (V_{abv},V_n)]	$\$ID_{E_{22}} \rightarrow \langle P_{108i}\rangle \rightarrow (\langle E_{12}\rangle \rightarrow P_{14} \rightarrow (E_{74} \rightarrow P_{01i} \rightarrow (PC_1 \rightarrow P_{02} \rightarrow \langle E_{41}\{=V_{abv}\}\rangle \rightarrow P_{1.1} \rightarrow \langle E_{55}\{\{\text{= 'Institution abbreviation'}\}\}\rangle) P_{01i} \rightarrow (PC_1 \rightarrow P_{02} \rightarrow \langle E_{41}\{=V_n\}\rangle \rightarrow P_{1.1} \rightarrow \langle E_{55}\{\text{= 'Institution name'}\}\rangle))$
16	['Producer Type', (T_y)]	$\$ID_{E_{22}} \rightarrow \langle P_{108i}\rangle \rightarrow (\langle E_{12}\rangle \rightarrow \langle P_{14}\rangle \rightarrow (\langle E_{74}\rangle \rightarrow P_2 \rightarrow \langle E_{55}\{=T_y\}\rangle)$
17	['Hierarchy',$(\text{Root}_{ref1},\text{Son}_{ref2})$]	$\|\|E_{31}\|\| \rightarrow \|\|P_1\|\| \rightarrow \|\|E_{42}\{=\text{Root}_{ref}\}\|\| \rightarrow P_{106} \rightarrow (\|\|E_{31}\|\| \rightarrow \|\|P_1\|\| \rightarrow (\|\|E_{42}\{=\text{Son}_{ref}\}\|\|)$
18	other rules	$\|\|E_{53}\|\| \rightarrow \|\|P_{89i}\|\| \rightarrow (\|\|E_{53}\|\| \rightarrow \|\|P_{55i}\|\| \rightarrow \$ID_{E_{22}} \rightarrow \|\|P_2\|\| \rightarrow \|\|E_{55}\{\text{= 'Original numbering'}\}\|\|) \rightarrow P_{89} \rightarrow (\|\|E_{53}\|\| \rightarrow \|\|P_2\|\| \rightarrow \|\|E_{55}\{\text{= 'Country'}\}\|\|\|\|P_{89i}\|\| \rightarrow \|\|E_{53}\|\| \rightarrow \|\|P_{55i}\|\| \rightarrow \$ID_{E_{22}})$

of 'E_{31} Document', that documents (property 'P_{70} documents') a physical object (an instance of 'E_{22} Human-Made Object) and refers to (property 'P_{67} refers to') a conceptual object (an instance of 'E_{33} Linguistic Object'). Therefore, the representation of the unit of description is translated into the rule No. 1 in Table 5. Figure 20(a) shows the representation of a unit of description in a diagram format layout.

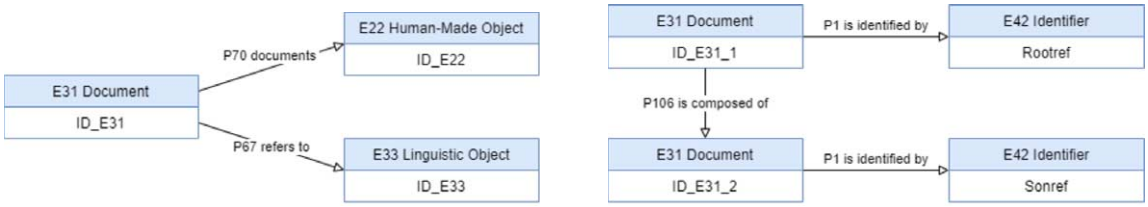
The hierarchical structure of the archives is represented using the relation 'P_{106} is composed of' between the 'E_{31} Document' and their sub-documents (also represented as individuals of the class 'E_{31} Document'). This representation is captured in rule No. 17, presented in Table 5 and in Fig. 20(b).

The Mapping Description Rules use the notation defined in [3] and widely used in the context of CIDOC-CRM mappings.

A mapping rule has a left and a right hand side. The left part is an attribute value term representing the information of the ISAD(G) element, the right part expresses the CIDOC-CRM representation. Variables for instances values, represented as a name ID, on the right part, are used to declare, ID, and refer to, $ID. Brackets are used to represent the label of a Class, for instance $E_{41}\{$'*Ana*'$\}$ represents that an instance of class E_{41} has label 'Ana'. The names of

Table 6
Mapping description language rules for the ontology knowledge discovery process

Rule No.	Left part (rec[attribute value list])	Right part CIDOC-CRM
19	['vessel',V]	$\$ID_{E_{31}} \rightarrow P_{129} \rightarrow (E_{22} \rightarrow P_2 \rightarrow (\langle E_{55}\{= V\}\rangle \rightarrow \langle P_2\rangle \rightarrow \langle E_{55}\{=$ `'vessel'`$\}))$
20	['products traded',V]	$\$ID_{E_{31}} \rightarrow P_{129} \rightarrow (E_{22} \rightarrow P_2 \rightarrow \langle E_{55}\{= V\}\rangle \rightarrow \langle P_2\rangle \rightarrow \langle E_{55}\{=$ `'products traded'`$\}))$
21	['Vessel name',V]	$\$ID_{E_{31}} \rightarrow P_{129} \rightarrow (E_{22} \rightarrow P_1 \rightarrow \langle E_{41}\{= V\}\rangle \rightarrow P_2 \rightarrow \langle E_{55}\{=$ `'vessel'`$\}))$
22	['Baptism', birth(Mother, Father, DBirth)]	$\$ID_{E_{31}} \rightarrow P_{67} \rightarrow (E_{67} \rightarrow P_{98} \rightarrow (\|E_{21}\| \rightarrow \|P_{02i}\| \rightarrow (\|PC_{129}\| \rightarrow \|P_{129.1}\| \rightarrow \|E_{55}\{$`'Recipient'`$\}\| \rightarrow \|P_{01}\| \rightarrow \$ID_{E_{31}})) \rightarrow P_{96} \rightarrow (E_{21} \rightarrow P_1 \rightarrow \langle E_{41}\{$`Mother`$\}\rangle) \rightarrow P_{97} \rightarrow (E_{21} \rightarrow P_1 \rightarrow \langle E_{41}\{$`Father`$\}\rangle) \rightarrow P_4 \rightarrow (E_{52} \rightarrow P_{170i} \rightarrow \langle E_{61}\{$`DBirth`$\}\rangle)$
23	['Baptism', bapt(Godfather, Godmother, DBap)]	$\$ID_{E_{31}} \rightarrow P_{129} \rightarrow (E_7 \rightarrow P_2 \rightarrow \langle E_{55}\{$`'Baptism'`$\}\rangle \rightarrow P_{01i} \rightarrow (PC_{14} \rightarrow P_{02} \rightarrow (\|E_{21}\| \rightarrow P_{02i} \rightarrow (\|PC_{129}\| \rightarrow \|P_{129.1}\| \rightarrow \|E_{55}\{$`'Recipient'`$\}\| \rightarrow \|P_{02}\| \rightarrow \$ID_{E_{31}})) \rightarrow P_{14.1} \rightarrow \langle E_{55}\{$`'Baptized'`$\}\rangle) \rightarrow P_{01i} \rightarrow (PC_{14} \rightarrow P_{02} \rightarrow (E_{21} \rightarrow P_1 \rightarrow \langle E_{41}\{$`Godmother`$\}\rangle) \rightarrow P_{14.1} \rightarrow \langle E_{55}\{$`'Godmother'`$\}\rangle) \rightarrow P_{01i} \rightarrow (PC_{14} \rightarrow P_{02} \rightarrow (E_{21} \rightarrow P_1 \rightarrow \langle E_{41}\{$`Godfather`$\}\rangle) \rightarrow P_{14.1} \rightarrow \langle E_{55}\{$`'Godfather'`$\}\rangle) \rightarrow P_4 \rightarrow (E_{52} \rightarrow P_{170i} \rightarrow \langle E_{61}\{$`DBap`$\}\rangle)$
24	['grandparents', mother(Mother, Father)]	$\$ID_{E_{31}} \rightarrow P_{67} \rightarrow (E_{67} \rightarrow P_{96} \rightarrow (E_{21} \rightarrow P_1 \rightarrow \langle E_{41}\{$`Mother`$\}\rangle) \rightarrow P_{97} \rightarrow (E_{21} \rightarrow P_1 \rightarrow \langle E_{41}\{$`Father`$\}\rangle) \rightarrow P_{98} \rightarrow (\|E_{21}\| \rightarrow \|P_{96i}\| \rightarrow (\|E_{67}\| \rightarrow \|P_{98i}\| \rightarrow (\|E_{21}\| \rightarrow \|P_{02i}\| \rightarrow (\|PC_{129}\| \rightarrow \|P_{129.1}\| \rightarrow \|E_{55}\{$`'Recipient'`$\}\| \rightarrow \|P_{02}\| \rightarrow \$ID_{E_{31}}))))$
25	['grandparents', father(Mother, Father)]	$\$ID_{E_{31}} \rightarrow P_{67} \rightarrow (E_{67} \rightarrow P_{96} \rightarrow (E_{21} \rightarrow P_1 \rightarrow \langle E_{41}\{$`Mother`$\}\rangle) \rightarrow P_{97} \rightarrow (E_{21} \rightarrow P_1 \rightarrow \langle E_{41}\{$`Father`$\}\rangle) \rightarrow P_{98} \rightarrow (\|E_{21}\| \rightarrow \|P_{97i}\| \rightarrow (\|E_{67}\| \rightarrow \|P_{98i}\| \rightarrow (\|E_{21}\| \rightarrow \|P_{02i}\| \rightarrow (\|PC_{129}\| \rightarrow \|PC_{129.1}\| \rightarrow \|E_{55}\{$`'Recipient'`$\}\| \rightarrow \|P_{02}\| \rightarrow \$ID_{E_{31}}))))$



(a) Rule No. 1 - Unit of Description (b) Rule No. 17 - Hierarchy

Fig. 20. Mapping description rules No. 1 and No. 17.

CIDOC-CRM classes or properties are abbreviated, for instance 'E_{41} Appellation' is represented as E_{41}. A right arrow, '$\rightarrow$', connects a class instance to a property or a property to a class instance, meaning that a new instance of the classes and of the property linking the class instances should be created and the result of the expression is the left hand side class instance.

The Mapping Description Language has been extended with the use of:

- $\langle$`expression`$\rangle$, where the 'expression' is added to the knowledge base only if the 'expression query' does not succeed. For instance, the expression query $ID_E \rightarrow \langle P_P\rangle \rightarrow \langle E_E\rangle$ will add the property P_P between the instance ID_E and a new instance of E_E if there is no property P_P between instance ID_E and any instance of E_E.
- $\|$`expression`$\|$, where 'expression' is ignored if the 'expression query' is not part of the knowledge base. For instance, the expression query $\|ID_{E_1}\| \rightarrow \|P_P\| \rightarrow \|ID_{E_2}\|$ will only be considered if the property P_P already exists between the existing instances ID_{E_1} and ID_{E_1}, otherwise is ignored and nothing is added to the knowledge base.

This expressiveness is important for the automatic interpretation of the mapping description rules that will allow to obtain the representation of the Portuguese National Archives in CIDOC-CRM representation. Each rule of the Table 5 gives rise to a sequence of assertions. These assertions are OWL2 facts in CIDOC-CRM.

In order to establish the adequate sequence of assertions for each rule, the expressiveness of the rules takes into account the following commands:

NewInst(I_{Id}, C_{Id}) – Creates a new instance of C_{Id} with value I_{Id}. If the instance already exists, raises an exception.

Inst(I_{Id}, C_{Id}, Value) – If there is an instance of C_{Id}, I_{Id} = Value; else creates a new instance I_{Id} of C_{Id} with value Value.

InstS(I_{Id}, C_{Id}, Value) – If there is an instance of the C_{Id}, I_{Id} = Value; else raises an exception.

NewProp(I_{Id_1}, P_{Id}, I_{Id_2}) – Creates a new instance of a object property P_{Id} with domain I_{Id_1} and range I_{Id_2}. I_{Id_i} are class instances. If the data property instance already exists raises an exception.

Prop(I_{Id_1}, C_{Id_1}, P_{Id}, I_{Id_2}, C_{Id_2}) – If there is an instance of an object property P_{Id} with domain I_{Id_1}, instance of the class C_{Id_1}, and range I_{Id_2}, instance of the class C_{Id_2}; else creates a new instance of the object property P_{Id} with domain I_{Id_1} and range I_{Id_2}.

PropS(I_{Id_1}, C_{Id_1}, P_{Id}, I_{Id_2}, C_{Id_2}) – If there is an instance of an object property P_{Id} with domain I_{Id_1}, instance of the class C_{Id_1}, and range I_{Id_2}, instance of the class C_{Id_2}; else raises an exception.

Therefore, rule No. 1 will be translated into the following sequence of commands:

$$
\begin{array}{ll}
1 & \text{NewInst}(ID_{E_{31}}, \text{`}E_{31}\text{ Document'}) \\
2 & \text{NewInst}(ID_{E_{22}}, \text{`}E_{22}\text{ Man-Made Object'}) \\
3 & \text{NewInst}(ID_{E_{33}}, \text{`}E_{33}\text{ Linguistic Object'}) \\
4 & \text{NewProp}(ID_{E_{31}}, \text{`}P_{70}\text{ documents'}, ID_{E_{22}}) \\
5 & \text{NewProp}(ID_{E_{31}}, \text{`}P_{67}\text{ refers to'}, ID_{E_{33}})
\end{array}
\tag{1}
$$

With a similar interpretation, rule No. 2, presented in Table 5 and Fig. 21(a), is translated into the following commands sequence:

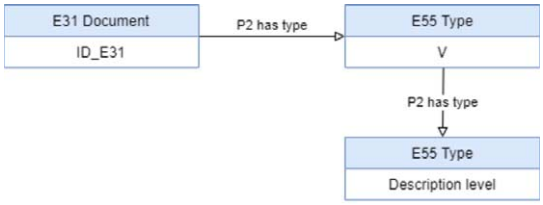
$$
\begin{array}{ll}
1 & \text{Inst}(ID_{E_{551}}, \text{`}E_{55}\text{ Type'},\text{V}) \\
2 & \text{Inst}(ID_{E_{552}}, \text{`}E_{55}\text{ Type'}, \text{`Description level'}) \\
3 & \text{Prop}(ID_{E_{551}}, \text{`}E_{55}\text{ Type'}, \text{`}P_2\text{ has type'}, ID_{E_{552}}, \text{`}E_{55}\text{ Type'}) \\
4 & \text{NewProp}(ID_{E_{31}}, \text{`}P_2\text{ has type'}, ID_{E_{551}})
\end{array}
\tag{2}
$$

Each commands sequence rule is then translated directly into Java instructions using the `OWL API` library (further details in Section 4.1).

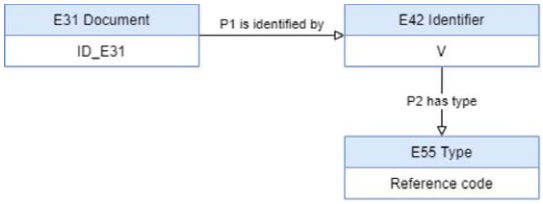
The Mapping Description Rule No. 18 (presented in Table 5 and Fig. 22(c)) states that the 'original numbering' place of a human-made object falls within the country place of the object unit. This rule must be triggered after the original numbering and country places were generated.

The rule No. 3, presented in Table 5 and displayed in Fig. 21(b), represents the 'Reference code' element interpretation in CIDOC-CRM.

Rule No. 4, in Table 5 and displayed in Fig. 22(a), represents the 'Language of material' interpretation in CIDOC-CRM.

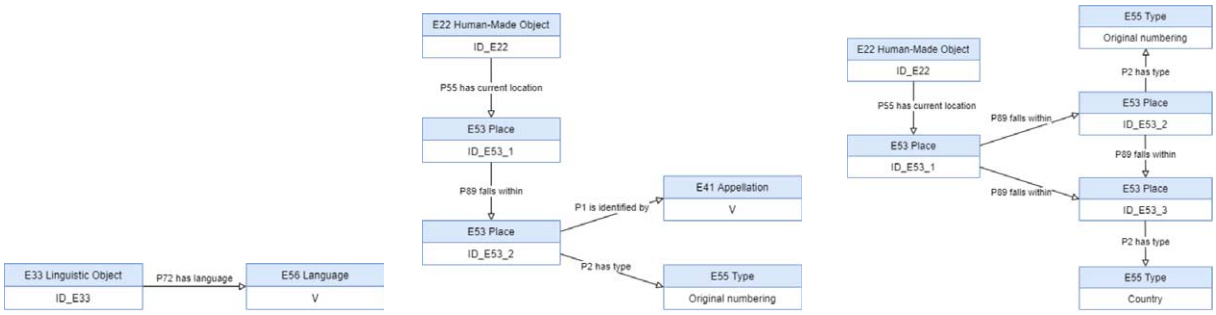


(a) Rule No. 2 - Description level



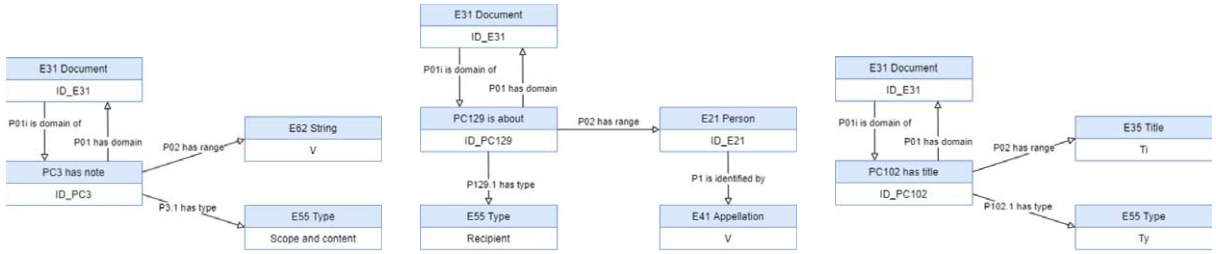
(b) Rule No. 3 - Reference code

Fig. 21. Mapping description rules No. 2 and No. 3.



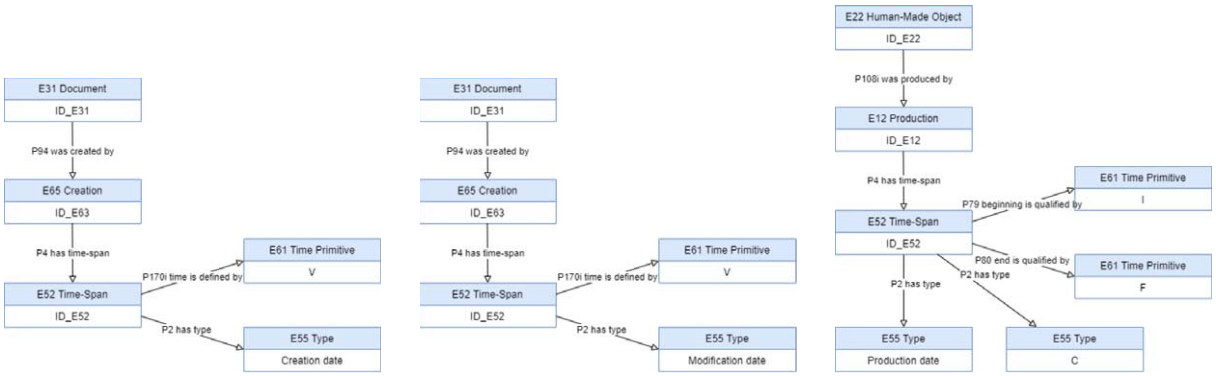
(a) Rule No. 4 - Language of material (b) Rule No. 5 - Original numbering (c) Rule No. 18 - Other Rules

Fig. 22. Mapping description rules No. 4, No. 5 and No. 18.



(a) Rule No. 6 - Scope and content (b) Rule No. 7 - Recipient (c) Rule No. 8 - Title and type

Fig. 23. Mapping description rules No. 6, No. 7 and No. 8.



(a) Rule No. 9 - Creation date (b) Rule No. 10 - Modification date (c) Rule No. 11 - Date range

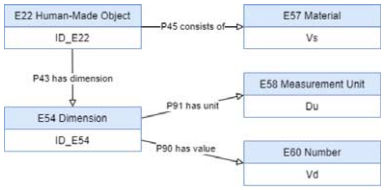
Fig. 24. Mapping description rules No. 9, No. 10 and No. 11.

For representing the ISAD(G) element 'Original numbering' Rule No 5, in Table 5, is used and is displayed in Fig. 22(b).

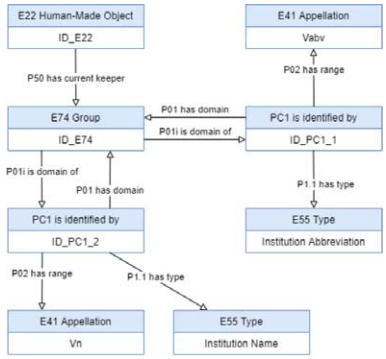
The representation of the element 'Scope and content', as well as other ISAD(G) text elements, is defined by Rule No. 6 in Table 5, and displayed in Fig. 23(a) to 26(b).

The 'Recipient' of an unit of description has the CIDOC-CRM representation stated by the rule No. 7, presented in Table 5 and displayed in Fig. 23(b).

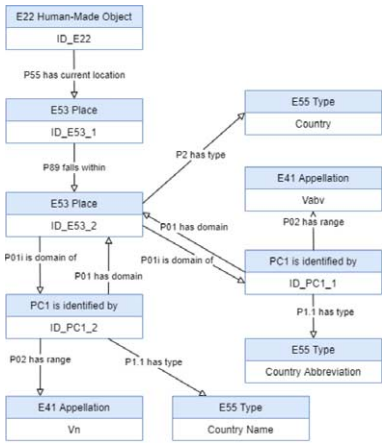
Rule No. 8, in Table 5 and Fig. 23(c), defines the representation of the elements 'Title' and 'Title type' in CIDOC-CRM.



(a) Rule No. 12 - Dimension and support

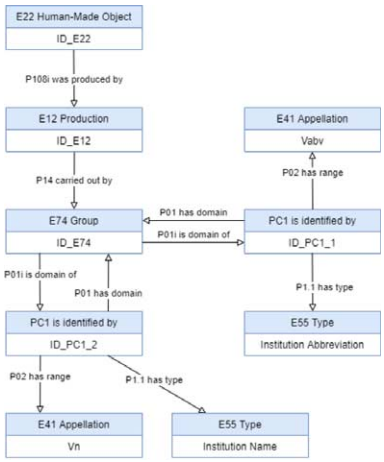


(b) Rule No. 13 - Current keeper

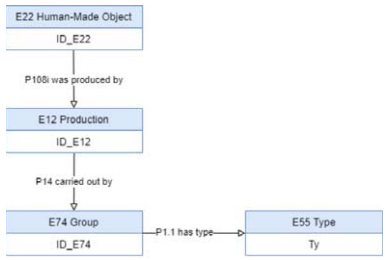


(c) Rule No. 14 - Country

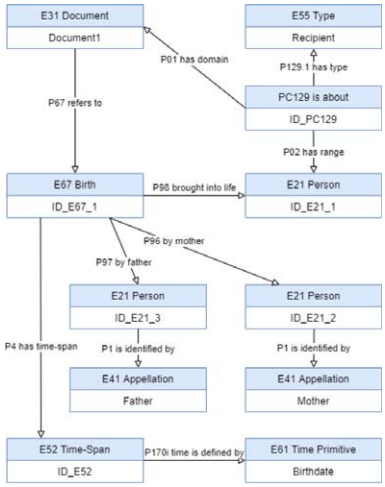
Fig. 25. Mapping description rules No. 12, No. 13 and No. 14.



(a) Rule No. 15 - Producer

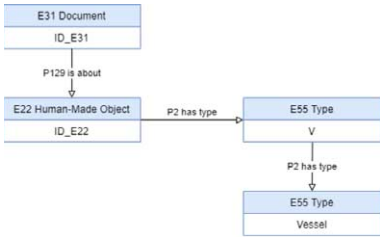


(b) Rule No. 16 - Producer Type

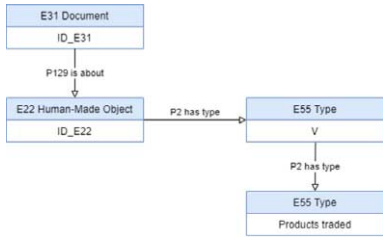


(c) Rule No. 22 - Birth Event of the Recipient, of a Baptism unit

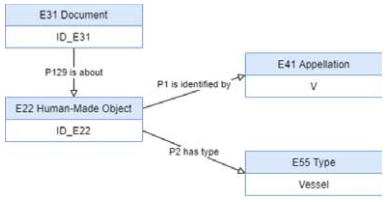
Fig. 26. Mapping description rules No. 15, No. 16 and No. 22.



(a) Rule No. 19 - Vessel

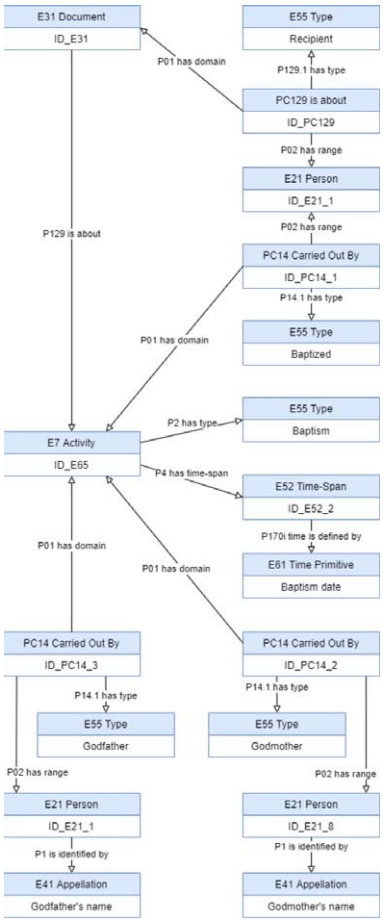


(b) Rule No. 20 - Products traded

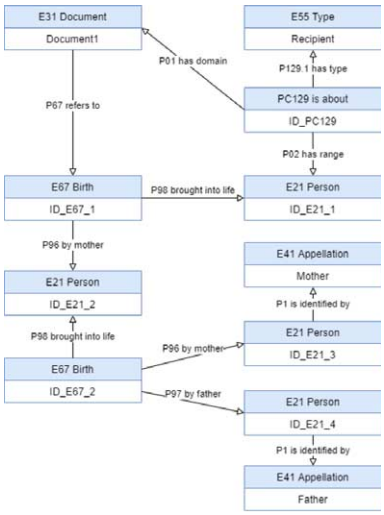


(c) Rule No. 21 - Vessel name

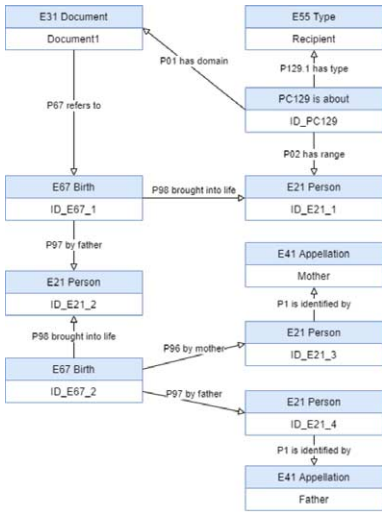
Fig. 27. Mapping description rules No. 18, No. 19 and No. 20.



(a) Rule No. 23 - Godparents of the Recipient, of a Baptism unit



(b) Rule No. 24 - Grandparents from mother's side of the Recipient, of a Baptism unit



(c) Rule No. 25 - Grandparents from father's side of the Recipient, of a Baptism unit

Fig. 28. Mapping description rules No. 23, No. 24 and No. 25.

The 'Creation Date' of an unit of description has the CIDOC-CRM representation defined by the rule No. 9, presented in Table 5 and displayed in Fig. 24(a).

Rule No. 10, in Table 5 and Fig. 24(b), defines the representation of the element 'Modification date' in CIDOC-CRM.

The 'Date Range' representation is defined by Rule No. 11 presented in Table 5, and displayed in Fig. 24(c).

The rule No. 12, presented in Table 5 and displayed in Fig. 25(a), represents the 'Dimension and support' of the physical object of the unit of description in CIDOC-CRM.

Rules No. 13 and No. 14, presented in Table 5 and displayed in Figs 25(b) and 25(c), are used to explicitly represent, respectively, the current keeper and the current location country of the physical object described in a unit of description.

The rules No. 15 and No. 16, presented in Table 5 and displayed in Figs 26(a) and 26(b), capture the representation of the 'Creator' of the unit of description and its type.

The rules, from No. 19 to No. 21, presented in Table 6 and displayed in Figures 27(a) to 28(c), are related to the representation of some pieces of text in CIDOC-CRM, introduced in Section 4.3. These representation concern to the representation of the concept 'Vessel' and its name, the concept 'Products traded', and connect them to the physical object of the unit of description.

Table 7
Mapping description rules for the RIC-O

Rule No.	Left part (rec[attribute value list])	Right part CIDOC-CRM	Right part RIC-O
1	DigitArq(Rec)	$E_{31}\{= ID_{E_{31}}\} \rightarrow P_{70} \rightarrow E_{22}\{= ID_{E_{22}}\} \rightarrow P_{67} \rightarrow E_{33}\{= ID_{E_{33}}\}$	$RecordSet\{= ID\_RS\} \rightarrow hasInstantiation \rightarrow Instantiation\{= ID\_I01\}$
2	[‘Description level’, V]	$\$ID_{E_{31}} \rightarrow P_2 \rightarrow (\langle E_{55}\{= V\}\rangle \rightarrow \langle P_2\rangle \rightarrow \langle E_{55}\{\texttt{= 'Description level'}\}\rangle)$	$\$ID\_RS \rightarrow hasRecordSetType \rightarrow \langle RecordSetType\{= V\}\rangle$
3	[‘Reference code’, V]	$\$ID_{E_{31}} \rightarrow P_1 \rightarrow (E_{42}\{= V\} \rightarrow P_2 \rightarrow \langle E_{55}\{\texttt{= 'Reference code'}\}\rangle)$	$\$ID\_RS \rightarrow hasOrHadIdentifier \rightarrow Identifier\{= V\}$
4	[‘Language of the material’, V]	$\$ID_{E_{33}} \rightarrow P_{72} \rightarrow \langle E_{56}\{= V\}\rangle$	$\$ID\_RS \rightarrow hasOrHadLanguage \rightarrow Language\{= V\}$
17	[‘Hierarchy’,($Root_{ref1}$,Son_{ref2})]	$\|\|E_{31}\|\| \rightarrow \|\|P_1\|\| \rightarrow \|\|E_{42}\{= \texttt{Root}_{ref}\}\|\| \rightarrow P_{106} \rightarrow (\|\|E_{31}\|\| \rightarrow \|\|P_1\|\| \rightarrow (\|\|E_{42}\{= \texttt{Son}_{ref}\}\|\|)$	$\|\|RecordSet\|\| \rightarrow \|\|hasOrHadIdentifier\|\| \rightarrow \|\|Identifier\{= \texttt{Root}_{ref}\}\|\| \rightarrow includesOrIncludes \rightarrow (\|\|RecordSet\|\| \rightarrow \|\|hasOrHadIdentifier\|\| \rightarrow (\|\|Identifier\{= \texttt{Son}_{ref}\}\|\|)$

Rule No. 22 defines the representation of a ‘Birth’ event and connects it to the Recipient of the description unit, see Table 6 and Fig. 26(c).

Rules No. 23 to No. 25 define the representation of the Godparents, Grandparents from mother and father side of the recipient of the description unit, see Table 6 and Figs 28(a), 28(b) and 28(c).

A.2. Mapping description rules for the RIC-O

This Subsection presents the set of rules, introduced in Section 4.5, to map ISAD(G) into RIC-O and following the proposed strategy representation. Table 7 presents these rules in a formal way by using the Mapping Description Language.

The figures for each of this rule are in Section 4.5.

References

[1] F. Beretta, A challenge for historical research: Making data FAIR using a collaborative ontology management environment (OntoME), *Semantic Web* (2020), 1–16.

[2] F. Beretta and V. Alamercery, Workflow for communal ontology management: Aligning data models with OntoME, in: *APOLLONIS Workshop «Historical Content Metadata»*, 2019.

[3] L. Bountouri and M. Gergatsoulis, The semantic mapping of archival metadata to the CIDOC CRM ontology, *Journal of Archival Organization* **9**(3–4) (2011), 174–207. doi:10.1080/15332748.2011.650124.

[4] M.C. Cavalcanti, F.D. Pereira, E. Fusco and M.L. Mucheroni, Model of data extraction in the innovation environments of the state of São Paulo based on semantic technologies, in: *CONTECSI – 14th International Conference on Information Systems & Technology Management*, CONTECSI USP, São Paulo, Brazil, 2017, ISSN 2448-1041. doi:10.5748/9788599693131-14CONTECSI/PS-4762.

[5] Definition of the CIDOC conceptual reference model, 7.0.1 edn, ICOM/CIDOC-CRM Special Interest Group, 2020.

[6] M.P. di Buono, M. Monteleone and A. Elia, How to populate ontologies, in: *Natural Language Processing and Information Systems*, E. Métais, M. Roche and M. Teisseire, eds, Springer International Publishing, Cham, 2014, pp. 55–58.

[7] M. Doerr, The CIDOC conceptual reference module: An ontological approach to semantic interoperability of metadata, *AI Magazine* **24**(3) (2003), 75, https://www.aaai.org/ojs/index.php/aimagazine/article/view/1720. doi:10.1609/aimag.v24i3.1720.

[8] Encoded archival description tag library, Version 2002, The Technical Subcommittee for Encoded Archival Description of the Society of American Archivists, 2002.

[9] P. Fragkou, N. Kritikos and E. Galiotou, Querying Greek governmental site using SPARQL, in: *Proceedings of the 20th Pan-Hellenic Conference on Informatics, PCI '16*, Association for Computing Machinery, New York, NY, USA, 2016. ISBN 9781450347891. doi:10.1145/3003733.3003807.

[10] S. Hennicke, Representation of archival user needs using CIDOC CRM, in: *Conference Proceedings TPDL: International Conference on Theory and Practice of Digital Libraries, Selected Workshops*, Valletta, Malta, 2013.

[11] M. Horridge and S. Bechhofer, The OWL API: A Java API for OWL ontologies, *Semantic Web* **2**(1) (2011), 11–21. doi:10.3233/SW-2011-0025.

[12] ISAD(G): General international standard archival description, Second Edition, International Council on Archives, 2011.

[13] P. Kremen and E. Sirin, SPARQL-DL implementation experience, in: *Proceedings of the 4th OWLED Workshop on OWL: Experiences and Directions Washington*, CEUR Workshop Proceedings, Vol. 496, 2007, CEUR-WS.org.
[14] I. Lourdi, C. Papatheodorou and M. Doerr, Semantic integration of collection description: Combining CIDOC-CRM and Dublin core collections application profile, *D-lib Magazine – DLIB* **15** (2009). doi:10.1045/july2009-papatheodorou.
[15] J. Makki, OntoPRiMa: A prototype for automating ontology population, *International Journal of Web/Semantic Technology (IJWesT)* **8** (2017).
[16] J. Makki, A.-M. Alquier and V. Prince, An NLP-based ontology population for a risk management generic structure, in: *Proceedings of the 5th International Conference on Soft Computing as Transdisciplinary Science and Technology, CSTST '08*, Association for Computing Machinery, New York, NY, USA, 2008, pp. 350–355. ISBN 9781605580463. doi:10.1145/1456223.1456296.
[17] O. Marlet, T. Francart, B. Markhoff and X. Rodier, OpenArchaeo for usable semantic interoperability, in: *ODOCH 2019 @CAiSE 2019*, Rome, Italy, 2019, https://hal.archives-ouvertes.fr/hal-02389929.
[18] D. Maynard, Y. Li and W. Peters, NLP techniques for term extraction and ontology population, 2008.
[19] C. Meghini and M. Doerr, A first-order logic expression of the CIDOC conceptual reference model, *International Journal of Metadata, Semantics and Ontologies* **13**(2) (2018), 131–149. doi:10.1504/IJMSO.2018.098393.
[20] D. Melo, I.P. Rodrigues and I. Koch, Knowledge discovery from ISAD, digital archive data, into ArchOnto, a CIDOC-CRM based linked model, in: *Proceedings of the 12th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management – Volume 2*, KEOD, SciTePress, 2020, pp. 197–204, INSTICC. ISBN 978-989-758-474-9. doi:10.5220/0010134101970204.
[21] D. Melo, I.P. Rodrigues and V.B. Nogueira, Using a dialogue manager to improve semantic web search, *International Journal on Semantic Web and Information Systems (IJSWIS)* **12**(1) (2016). doi:10.4018/IJSWIS.2016010104.
[22] D. Melo, I.P. Rodrigues and V.B. Nogueira, Semantic web search through natural language dialogues, in: *Innovations, Developments, and Applications of Semantic Web and Information Systems*, D.L. Miltiadis, N. Aljohani, E. Damiani and K.T. Chui, eds, IGI Global, Hershey, 2018, pp. 329–349. doi:10.4018/978-1-5225-5042-6.ch012.
[23] D. Melo, I.P. Rodrigues and D. Varagnolo, Semantic migration of fonds examples, *Mendeley Data* **V1** (2020). doi:10.17632/knx84z3463.1.
[24] D. Melo, I.P. Rodrigues and D. Varagnolo, Installation unit – REGISTOS DE BAPTISMOS, *Mendeley Data* **V2** (2020). doi:10.17632/wx7v7rmg7h.2.
[25] D. Metilli, V. Bartalesi and C. Meghini, A Wikidata-based tool for building and visualising narratives, *International Journal on Digital Libraries* **20** (2019), 417–432. doi:10.1007/s00799-019-00266-3.
[26] D. Myers, M.S. Quintero, A. Dalgity and I. Avramides, The Arches heritage inventory and management system: A platform for the heritage field, *Journal of Cultural Heritage Management and Sustainable Development* (2016).
[27] Orientações para a Descrição Arquivística, 3rd edn, Direção-Geral de Arquivos, Grupo de Trabalho de Normalização da Descrição, Ministério da Cultura, Lisbon, Portugal, 2011.
[28] D. Pitti, B. Stockting and F. Clavaud, An introduction to "Records in Contexts": An archival description draft standard, *Comma* **2016**(1–2) (2018), 173–189. doi:10.3828/comma.2016.18.
[29] J.C. Ramalho and J.C. Ferreira, DigitArq: Creating and managing a digital archive, in: *Building Digital Bridges: Linking Cultures, Commerce and Science: 8th ICCC/IFIP International Conference on Electronic Publishing Held in Brasília – ELPUB 2004*, Brasilia, Brazil, June 23–26, 2004, Proceedings, 2004.
[30] E. Sirin and B. Parsia, SPARQL-DL: SPARQL query for OWL-DL, in: *3rd Workshop on OWL: Experiences and Directions*, CEUR Workshop Proceedings, Vol. 258, 2007, CEUR-WS.org.
[31] M. Theodoridou, G. Bruseker, M. Daskalaki and M. Doerr, Methodological tips for mappings to CIDOC CRM, in: *Proceedings of CAA 2016 – 44th Computer Applications and Quantitative Methods in Archaeology Conference, CAA 2016 OSLO Exploring Oceans of Data*, 29 March–2 April, 2016.
[32] M. Theodoridou and M. Doerr, Mapping of the encoded archival description DTD element set to the CIDOC CRM, Technical report FORTHICS/TR-289, FORTH, 2001.
[33] M. Theodoridou and M. Doerr, Modelling properties of properties, 2016, http://new.cidoc-crm.org/Resources/modeling-properties-of-properties-in-the-cidoc-crm-rdf-encoding.
[34] S. Vitali, Authority control of creators and the second edition of ISAAR (CPF), international standard archival authority record for corporate bodies, persons, and families, *Cataloging & classification quarterly* **38**(3–4) (2004), 185–199. doi:10.1300/J104v38n03_15.