Semantic Web 0 (0) 1
IOS Press

Pattern-based design applied to cultural heritage knowledge graphs

ArCo: the knowledge graph of Italian Cultural Heritage

Valentina Anita Carriero [a,*], Aldo Gangemi [b,d], Maria Letizia Mancinelli [c], Andrea Giovanni Nuzzolese [d], Valentina Presutti [e,d] and Chiara Veninata [c]
[a] *DISI, University of Bologna, Italy*
E-mail: valentina.carriero3@unibo.it
[b] *FICLIT, University of Bologna, Italy*
E-mail: aldo.gangemi@unibo.it
[c] *ICCD, MiBAC, Italy*
E-mails: marialetizia.mancinelli@beniculturali.it, chiara.veninata@beniculturali.it
[d] *STLab, ISTC-CNR, Italy*
E-mail: andreagiovanni.nuzzolese@cnr.it
[e] *LILEC, University of Bologna, Italy*
E-mail: valentina.presutti@unibo.it

Abstract. Ontology Design Patterns (ODPs) have become an established and recognised practice for guaranteeing good quality ontology engineering. There are several ODP repositories where ODPs are shared as well as ontology design methodologies recommending their reuse. Performing rigorous testing is recommended as well for supporting ontology maintenance and validating the resulting resource against its motivating requirements. Nevertheless, it is less than straightforward to find guidelines on *how* to apply such methodologies for developing domain-specific knowledge graphs. ArCo is the knowledge graph of Italian Cultural Heritage and has been developed by using eXtreme Design (XD), an ODP- and test-driven methodology. During its development, XD has been adapted to the need of the CH domain e.g. gathering requirements from an open, diverse community of consumers, a new ODP has been defined and many have been specialised to address specific CH requirements. This paper presents ArCo and describes *how* to apply XD to the development and validation of a CH knowledge graph, also detailing the (intellectual) process implemented for matching the encountered modelling problems to ODPs. Relevant contributions also include a novel web tool for supporting unit-testing of knowledge graphs, a rigorous evaluation of ArCo, and a discussion of methodological lessons learned during ArCo development.

Keywords: cultural heritage knowledge graph, ontology design patterns, ontology testing, ontology evaluation

1. Introduction

Museums, libraries, archives, private collections and other cultural institutions have the essential mission to preserve the cultural objects they collect. Hence, data about these objects is of utmost importance, since it allows to keep memory of them, their life cycle as well as their artistic, social, and historical context. If data are shared, they can be used as a means of enhancing cultural properties, by spreading knowledge on cultural heritage, and widening its potential consumers. Cultural Heritage (CH) data can have various types of consumers such as citizens, students, scholars, scientists, managers, public administrations, and companies. Consequently, it can impact on different domains such as tourism, research, management, teaching, etc. Moreover, cultural institutions can mutually

*Corresponding author. E-mails: valentina.carriero3@unibo.it, valentina.presutti@unibo.it.

benefit from the data they publish, especially by creating connections between their knowledge bases. The Linked Data paradigm has shown its effectiveness in supporting this practice [1], and its adoption in the Cultural Heritage domain is leading to a significant transformation in the management of CH data [2–6].

The Italian Cultural Heritage is an important part of the whole world CH[1], and a great resource for Italy from aesthetic, social, historical, cognitive, and economic points of view. More and more cultural institutions are publishing their data, often as open data, in order to allow for interchange, interlinking and mutual enrichment.

In [7] we introduce ArCo, a recent resource that contributes to this vision by publishing a knowledge graph consisting of a network of ontologies modelling the CH domain, a LOD dataset of ∼169M triples about Italian cultural properties, along with data, documentation and software artefacts produced along. The semantics emerging from ArCo addresses directly the CH domain, by distinguishing knowledge of cultural entities and their context, versus the knowledge dynamically assembled in catalographic records. The first type creates a CH ontology unprecedented in its depth and latitude, while the second allows to trace the epistemological aspects of CH catalogues.

For this reason, besides the relevance of the produced resource, addressed in [7], ArCo as a project can contribute to push the state of the art in knowledge graph engineering, with special focus on the CH domain, by sharing its "behind the scene", i.e. the intellectual and methodological processes performed, the adopted design principles, and the lessons learned, which constitute the main focus of this paper.

ArCo is an evolving creature, both as a knowledge graph, and as a methodology. New requirements are continuously collected, incremental versions are regularly released, and its methodological approach is discussed with the community, and possibly refined and evolved[2].

ArCo knowledge graph derives from the General Catalogue of Italian Cultural Heritage[3] (GC), which is maintained by the Central Institute for Catalogue and Documentation (ICCD) of the Ministry of Cultural Heritage and Activities[4] (MiBAC). ICCD coordinates the cataloguing activities by collecting and integrating data coming from diverse institutions all over Italy with the help of a collaborative platform named SIGECweb[5]. The General Catalogue data are finally stored in a relational database (and encoded in XML). In order to convert such data into a knowledge graph, a reference ontology model that captures the concepts expressed in the database is required.

There are several good reference models for representing Cultural Heritage data and publishing them as linked open data. The Europeana Data Model (EDM) [5] and CIDOC Conceptual Reference Model (CRM) [8] are two prominent examples. However, their ontological commitment seems focused on allowing a linked data encoding of the metadata that can be extracted from typical catalogue records. Although ArCo data source is a set of catalogue records, the ambition of the project is larger than transforming them into linked data. ArCo aims at modelling the entities that populate the Cultural Heritage domain, the events they participate in, the types of places they are located in, the processes they are involved in, etc., hence providing the CH and the Semantic Web communities with a set of ontology patterns to represent the cultural entities' knowledge domain. ArCo's ultimate goal is to enable researchers and scholars to make new findings about cultural entities, and to develop new theories based on observations performed on CH knowledge graphs modelled by means of its ontologies.

ArCo ontologies are aligned to EDM and CIDOC-CRM in order to facilitate linking and reuse by aggregators, but they are much richer and extended in the variety and granularity of concepts they provide. As an example, the painting "Woman Portrait" by Netscher Caspar (17th century) is associated with several types of locations: it is now located at the Uffizi in Florence, it was stored in 1942 at Poppi Castle, it was involved (hence temporarily moved) in an exhibition at Pitti Palace in Florence in 1773 (cf. Figure 1).

CIDOC-CRM allows us to encode the data about all these locations by means of a "move" event that is both temporally and geographically indexed. This represen-

[1]According to UNESCO, Italy is the country with the highest heritage sites in the world, https://doi.org/10.1007/978-3-030-30796-7_3

[2]ArCo releases and issue tracker can be found at https://github.com/ICCD-MiBACT/ArCo, here you can download ArCo as a docker and have everything on your own PC. Additional documentation, user guide, and examples can be found at https://w3id.org/arco. ArCo methodology is discussed on a dedicated mailing list arco-project@googlegroups.com as well as during webinars and meetups

[3]http://www.catalogo.beniculturali.it

[4]www.beniculturali.it

[5]http://www.iccd.beniculturali.it/it/sigec-web

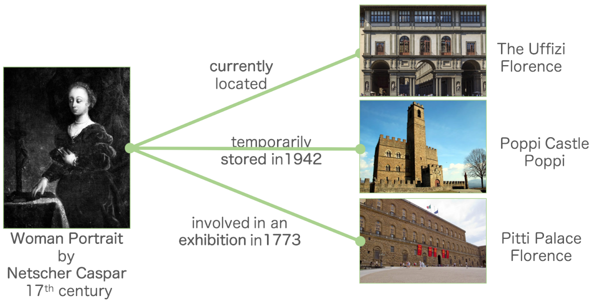


Fig. 1. The painting “Woman Portrait” by Netscher Caspar (17th century) and the different (types of) locations it is associated with.

tation keeps off the information about the type or motivation of a specific location, i.e. production, exhibition, storage, etc., as well as the temporal validity of that location, which is different from the time at which a moving event occurs. In turn, location types can be further characterised by other data or entities specific for them. In ArCo it is possible to both represent the moving event with its temporal and spatial indexing, and the type of location with its own temporal validity.

Other examples include: the modelling of the catalogue records as entities of the CH domain representing the epistemological perspective on cultural properties as opposed to their ontological perspective, the recurrence of cultural events such as periodic festivals, exhibitions, etc., as well as other situations in which cultural properties are involved: observations, issuances, etc.

In order to govern a knowledge graph development process able to address requirements from a diversity of potential consumers, to provide a rich expressivity, and to preserve high quality and easiness of reuse, ArCo follows a pattern-based ontology design methodology named eXtreme Design (XD) [9, 10], and has contributed to extend and improve it.

Contribution This paper significantly extends [7] by providing further details on ArCo knowledge graph (including examples) and its development context. Additionally, its novel contributions can be summarised as follows:

- it describes how to implement the eXtreme Design methodology for a CH project, by pointing out what tools can be used for the various activities;
- it extends eXtreme Design by implementing an *open gathering* approach to requirements collection, which supports the variety of potential consumers of CH data;
- it describes the process of, and motivations for, key design choices, e.g. the modules composing the ontology network, the model redundancy by representing same concepts as both *n*-ary and binary relations, the direct versus indirect reuse of existing ontologies;
- it describes the intellectual process that leads to mapping modelling issues to specific ontology design patterns for three key concrete and prominent cases in CH;
- it describes how to perform ontology testing, and provides a novel web tool for running regression tests on a knowledge graph;
- it formally evaluates ArCo ontology network according to a set of established metrics [11–17];
- it tells the lessons learned during the development process.

The paper is structured as follows. Section 2 describes the General Catalogue of Italian Cultural Heritage. Section 3 discusses relevant related work, with regard to both LOD, ontologies and methodologies for CH data. Section 4 shows how we applied eXtreme Design principles to ArCo, for modelling the Cultural Heritage domain. Section 5 provides examples of usage of ArCo. Section 6 contains a rigorous evaluation of the presented knowledge graph. In Section 7 the lessons learned during ArCo development are presented. Finally, Section 8 concludes the paper with ongoing and future works.

2. The journey through semi-structured data on Italian Cultural Heritage

Building a knowledge graph and its reference ontology network requires to understand the domain and the ontological commitment that its conceptualisation conveys, and to transform the available data into linked entities that comply with the resulting ontologies. There may be different scenarios in terms of what is available at the beginning of a knowledge graph project, but one of the most common situations is having a (set of) database(s) where the data are stored and maintained, and that are used as main sources for feeding some presentation interface. Along with a continuous interaction with the administrators of the databases and the domain experts, these resources are to be analysed in order to extract the (often implicit) conceptual model of the domain that they encode. ArCo’s main datasources have been an XML database of catalogue records and a set of pdf documents describing catalogue standards, encoded as XSD schemas.

Cataloguing cultural heritage is the process of identifying and describing, through metadata, information resources that are considered cultural properties, by virtue of their historic, artistic, archaeological and ethnoanthropological interest. In Italy, the Italian Ministry of Cultural Heritage and Activities[4] (MiBAC), regions and local agencies are in charge of cooperatively cataloguing Italian cultural heritage they own, aiming at safeguarding, enhancing, and making publicly available data on, cultural heritage.

2.1. The General Catalogue of Italian Cultural Heritage

ICCD coordinates these cataloguing activities by maintaining the General Catalogue of Italian Cultural Heritage[3] (GC), which is the official institutional database of Italian cultural heritage, promoting integrated management of data coming from all over Italy and from diverse institutions and local contexts.

The General Catalogue is built upon a collaborative platform, named SIGECweb[5], to which national or regional, public or private, institutional organisations that administer cultural properties can submit their catalogue records, i.e. files containing data on cultural properties and compliant with predetermined standards and guidelines (see Subsection 2.2). Only users from institutions that are formally authorised by ICCD can access and contribute to SIGECweb. Each user is associated with a specific profile (e.g. administrator, cataloguer), which determines the (sub-)set of actions and functionalities that can be executed through the platform. The entire process of creating a catalogue record is checked and validated, both automatically and by appointed experts. Thus, the quality of the database and its highly reliable provenance is guaranteed by (i) an accreditation process that allows only authorised entities to contribute to the platform, (ii) a data validation phase performed by heritage protection agencies that assess the scientific quality of catalogue records, and (iii) a data validation phase based on compliance with specific cataloguing standards (see Figure 2.

SIGECweb currently contains 2,735,343 catalogue records, 831,114 of which are publicly accessible through the General Catalogue. The remaining records are associated with a privacy level which prevents them to be openly published, since they refer to either private properties, or properties being at stake (e.g. items in unguarded buildings), or properties still requiring a scientific assessment by accounted institutions, etc.

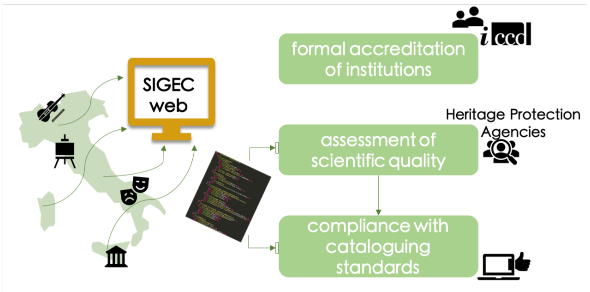


Fig. 2. Accreditation and validation process for contributing to SIGECweb.

2.2. Italian Cataloguing Standards

In order to guarantee high quality and interoperability between data accessible through the General Catalogue, ICCD defines a set of standards to allow for consistent cataloguing of various cultural properties, across different institutions throughout the Italian territory. The set of standards (*normative*)[6] for encoding catalogue records (*schede di catalogo*) provide a template for collecting and organising data on different types of cultural properties, and a methodological base for cataloguing. Thus, these standards are part of ArCo's input, along with data contained in the GC catalogue records.

Each cataloguing standard consists of a PDF document[7] that contains, as shown in Figure 3: a table listing all fields that can be filled for collecting data about a cultural property, and the respective rules for compilation. These fields are grouped into *paragraphs* with regard to the topic (e.g. geographical information), following a hierarchical structure. To each paragraph and field in the hierarchy are associated a tag and generic instructions: maximum number of characters allowed, whether that field can be filled more than once or not, whether its compilation is mandatory for considering the catalogue record valid or not, etc. Moreover, all fields are accompanied by detailed rules for compilation, such as syntactic rules (e.g. the date format), and useful examples.

[6] http://www.iccd.beniculturali.it/it/normative

[7] These standards are gradually being published also in XSD format.

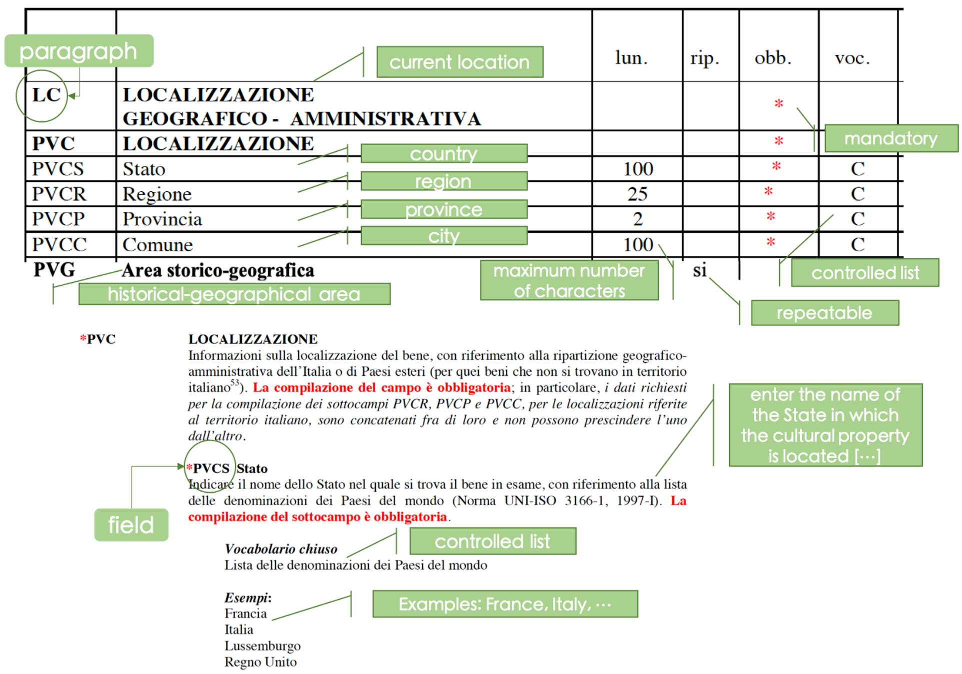


Fig. 3. An example of the structure of an ICCD cataloguing standard.

2.2.1. 30 types of cultural property, 30 cataloguing standards

ICCD collects catalogue records about 9 categories of cultural properties, which generalise over 30 different more specific types: archaeological, architectural and landscape, demo-ethno-anthropological, photographic, musical, natural, numismatic, scientific and technological, historical and artistic properties. For each of the 30 typologies, a specific cataloguing standard has been defined. Many parts of these standards are similar, and a recent effort to map these sections was made: in 2015 a *core* cataloguing standard (*Normativa Trasversale*) has been released, which groups and defines the fields common to all kinds of cultural property. Nevertheless, the particular features of each cultural property type require specific standards for defining additional paragraphs and fields. For instance, the standard for cataloguing musical instruments (SM) contains a paragraph dedicated to the description of possible accessories, such as the instrument case or the piano bench. Moreover, some fields are associated with controlled lists, i.e. lists of non-overlapping terms used to control terminology and indicate all the valid values for filling one field. In many cases, these controlled lists differ, partially or completely, depending on the cultural property that is being catalogued: for example, according to the standard for photographs, the list associated with the field *type of measurement* contains values such as "height x length" or "height x length x thickness", while, when cataloguing technological heritage, examples of valid values are "weight" and "volume".

Currently, an effort is being made by ICCD in publishing on GitHub[8] many of these controlled lists in RDF using SKOS.

2.2.2. One cataloguing standard, different versions over time

ICCD has been sharing cataloguing standards since 1990. These standards have undergone changes and updates, regarding both their structure and rules for compilation. Thus, each cultural property type has been associated with different versions[9] of catalogue standards over time: as a result, the GC contains heterogeneous catalogue records, following different versions, thus requiring expensive and time-consuming mapping activities: indeed, in moving from a version to the next one, ICCD did not keep track of changes,

[8] https://github.com/ICCD-MiBACT/Standard-catalografici/tree/master/strumenti-terminologici

[9] Previous and current versions include: 1.00 and 2.00 (1990-2000), 3.00 (2002-2004), 3.01 (2005-2010), 4.00 (since 2015).

systematically. While in some cases this mapping is straightforward, in other cases differences over data due to different versions can be significant. Let us consider an example, depicted in Figure 4, of the standard F (for cataloguing photographs). In version 3.00 there are three separate fields to indicate *place, site*, and *date* of a photograph (`MSTL`, `MSTD`, `MSTS`, respectively), while in version 4.00, they are all encoded in a single field (`MSTL`).

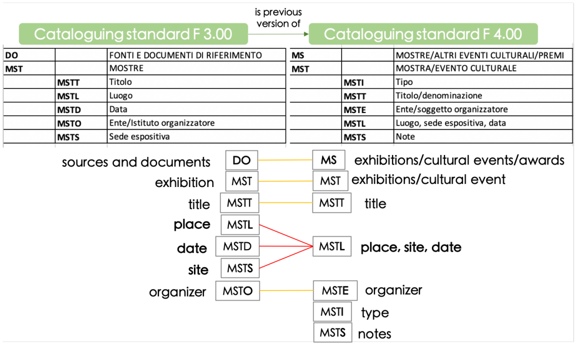


Fig. 4. The ICCD cataloguing standard *F* for photographs: difference and mapping between version 3.00 and version 4.00.

2.3. A closer look at actual catalogue records

Although catalogue records submitted to SIGECweb are subject to a validation process, being collaborative in nature means that catalogue records are not error-free. There are cases of: mandatory fields that are not properly filled, thus producing an error code in the final catalogue record version; catalogue records containing values alternative to those provided by controlled lists, hence undermining data homogeneity; use of non-standard formats (e.g. for dates); minor bugs and typing errors. ICCD is continuously working for improving the collecting process, in order to minimise these situations.

Moreover, catalogue standards themselves could be improved in their structure, in order to maximize data mining from catalogue records: there are still many fields allowing for long descriptive texts, from which structured high-quality information could be derived and extracted.

3. LOD, ontologies and methodologies in the cultural heritage domain

The Semantic Web and the LOD principles have changed how cultural institutions manage and publish their data, how machines and users can access linked and enriched data on Cultural Heritage (CH), and have widened the possibility of reuse and generation of new knowledge starting from existing data. Ontologies make it possible to go beyond traditional CH data production and publication, providing users with new, more intelligent and eventually personalised Web applications and services, and with more and richer data [4].

LOD and ontologies for Cultural Heritage. Projects such as LODLAM[10] and OpenGLAM[11] give evidence of a growing community interested in these themes. Many cultural institutions are now making cultural properties they safeguard accessible online, by releasing their datasets as Linked Open Data [18]. In the Italian context, we can cite as notable examples the Zeri&Lode[12] project, which publishes as LOD existing metadata collections of the Zeri Photo Archive [6], and LOD published by the Institute of Artistic, Cultural and Natural heritage of the region Emilia-Romagna[13] (IBC-ER), which include data on libraries and museums, historic castles, butterflies and monumental trees, etc. Other notable examples of Linked Data projects include the Rijksmuseum Amsterdam collection[14] [2], the Smithsonian Art Museum[15] [19], and the German National Library. A big effort is being made also in organising knowledge on CH through the publication of controlled vocabularies, as in the case of Getty Vocabularies[16], which contain structured terminology for art, architecture, archival and bibliographic materials (e.g. ULAN for artist names, TGN for places relevant to the CH domain, etc.).

Publishing and interconnecting data is leading to the creation of international CH portals [4], such as Europeana[17], Google Arts & Culture[18], and MuseumFinland [20], which aggregate content from various publishers into a single site as a point of access of heterogeneous collections; they are referred as *aggregators*.

Along with the publication of LOD collections, ontologies representing the CH domain are being modelled, and some of them are becoming widely adopted

[10]http://lodlam.net
[11]http://openglam.org/
[12]http://data.fondazionezeri.unibo.it/
[13]https://ibc.regione-emilia-romagna.it/servizi-online/lod/
[14]http://datahub.io/dataset/rijksmuseum
[15]http://americanart.si.edu/collections/search/lod/about/
[16]http://www.getty.edu/research/tools/vocabularies/index.html
[17]https://www.europeana.eu/portal/en
[18]https://artsandculture.google.com/

standards, in particular the Europeana Data Model[19] (EDM) [5] and CIDOC Conceptual Reference Model (CRM)[20] [8]. In addition to them, many other ontologies model specific domains that can be relevant and related to CH (e.g. PRO ontology[21] for agents' roles).

As explained in [18], EDM has been developed for integrating and making interoperable various metadata standards from a multitude of Galleries, Libraries, Archives and Museums (GLAM) across Europe, as a "common denominator" model to use for the Europeana portal.

Therefore, intentionally, it does not deepen in a high level of granularity or axiomatisation. It provides a basic set of classes and properties, which also include many Dublin Core[22] constructs, hence its ontological commitment is underspecified. EDM has an object-centric approach, where the cultural property is directly connected to its features, hence reducing the possibility to express temporal and contextual information. While such model can be sufficient for addressing the requirements of an aggregator of many different collections, it hardly answers to the need of a cultural institution aiming at giving full expressiveness to its data and at going beyond mere data management.

CIDOC-CRM is a richer model than EDM and has an event-centric approach, where many of the features expressed as object properties in EDM are modelled using an event, better capturing changes over time. Even in the case of CIDOC, many requirements led us to the need of new modelling effort. ArCo needs to satisfy a significant number of modelling issues, overlooked by other ontologies so far, such as the diagnosis of a paleopathology and the interpretation of sex and age of death in anthropological material, other types of surveys on cultural properties (e.g. laboratory tests), the coin issuance, the Hornbostel-Sachs classification of musical instruments, recurrent art exhibitions, etc. Moreover, in many cases CIDOC lacks the expressiveness needed for modelling ArCo data without loosing information. For example, according to CIDOC, changes of the physical location of a cultural property are represented by move events, and we can only know from and to where the cultural property was moved, while there is no means to express e.g. the location type or its temporal validity. CIDOC distinguishes, by means of relations between a cultural property and a place, 3 types of locations: current, current or former, current permanent. In order to satisfy our requirements in representing physical locations associated with a cultural property, we need more expressiveness, e.g. for distinguishing between different types of locations with a temporal validity: the place were a cultural property was exhibited, the place where it was found, etc. (see Subsection 4.3.1 for more details about ArCo ontology design choices associated with locations).

ArCo is aligned to both CIDOC-CRM and EDM, as well as to other ontologies such as BIBFRAME[23], FRBR[24], FaBiO[25] (for bibliographic data), FEntry[26] and OAEntry[27] (dedicated to photographs and artworks). The richness and high level of detail of ArCo requirements though led us to perform a consistent modelling effort and to release to the community a number of useful ontology patterns for representing the CH domain.

Methodologies for CH LOD modelling and publishing. As discussed in [18], when building a knowledge graph for publishing its data, a cultural institution makes a first relevant choice: it can either publish Linked Open Data by building and using its own infrastructure, or give its data to a cultural heritage data aggregator such as Europeana. A third case is of an institution that invests in infrastructure for publishing its data as well as in the whole process for producing them, by using the ontology model of an aggregator.

In making this choice, a cultural heritage administrator is influenced by different aspects both political, economical and technical. An aggregator provides a single point of access to different collections from many cultural institutions, giving visibility and guaranteeing respective enrichment and interoperability. Nevertheless, the adopted ontologies only capture a subset and a simplified encoding of the available information about a cultural property because they prefer a *lightweight* modelling i.e. based on binary relations, as opposed to more complex predicated, e.g. *n*-ary relations. Many existing CH institutions provide data to Europeana and/or use Europeana Data Model, along with Dublin Core, for representing their collections, such as the Rijksmuseum dataset [2], possibly extending it, as in the case of the VVV ontology [21]. In each

[19] https://pro.europeana.eu/page/edm-documentation
[20] http://www.cidoc-crm.org/
[21] http://purl.org/spar/pro/
[22] http://dublincore.org
[23] http://id.loc.gov/ontologies/bibframe.html
[24] http://vocab.org/frbr/core
[25] http://purl.org/spar/fabio
[26] http://www.essepuntato.it/2014/03/fentry
[27] http://purl.org/emmedi/oaentry

of these projects, the institution intentionally chooses to publish only a subset of all the features characterising cultural properties, which are instead present in the original dataset, in order to reuse EDM and avoid a more significant effort in mapping between the input data and the ontology model. In our opinion, when possible, it is preferable for an institution to carry out the whole process of data production and publication and to release as much rich data as possible, while guaranteeing the interoperability with and the publication (of simplified or subsets of its data) through aggregators: this is the approach followed by ArCo.

4. Applying eXtreme Design principles to model the Cultural Heritage domain

In order to cope with such a huge amount of conceptual entities as those covered by the General Catalogue and to ensure a high quality modular architecture of ArCo ontologies, we adopt an established pattern-based ontology methodology, named eXtreme Design (XD) [9, 10].

4.1. eXtreme Design methodology

After the *ontology project initiation*, each iteration of XD (depicted in Figure 5) involves: a *customer team*, which elicits, in the form of textual stories, the requirements that guide the design and testing processes; a *design team*, which is in charge of selecting and implementing the Ontology Design Patterns (ODPs) [22] that best address the given requirements. Requirements are derived by the customer stories: they are refined through iterative interactions between the design and customer teams and are transformed into competency questions (CQs) [23] and general constraint statements. The CQs identify the queries that we want to resolve against the knowledge graph. The general constraints are formalised as axioms. CQs and constraints represent the project's ontological commitment; a *testing team* which performs testing and validation of the produced ontology components against the CQs and general constraints, and feedback to the design team. If the validation results in any error or issue to be addressed, the design team will address them and resubmit to the testing team; an *integration team* which takes care of integrating the different components into the ontology network, and of ensuring a consistent and homogenous vocabulary. The results of the integration are also submitted to the testing team, and follow the same feedback cycle described before.

The key aspect of XD, and in general of pattern-based design, is the ability to *match* CQs to ODPs. Section 4.3 describes this process in detail by motivating why certain choices are made in ArCo. A fundamental part of the design process is the testing activity. XD is test-driven and follows a unit testing approach as described in [24]. In Section 4.5, we provide details on how ArCo KG has been tested, and which problems have been faced and addressed in performing this activity. Finally, Section 4.6 shows how XD has been adapted and implemented in order to handle an open gathering approach to requirements collection from a diverse audience of potential consumers, which is a characterising feature of CH data.

4.2. Identifying the modules of ArCo ontology network

The input material provided to ArCo *design team* consists of: (i) all versions of the reference cataloguing standards for 30 types of cultural properties; (ii) a *core* cataloguing standard, expressing common information across all reference standards; (iii) about 800,000 cataloguing records; (iv) a set of user stories elicited by the *customer team*, i.e. ICCD experts as well as external actors.

In agreement with ICCD domain experts, we started by analysing the *core* cataloguing standard (*Normativa Trasversale*) and related user stories, aiming at first addressing concepts that are common to all cultural properties. In a later phase, the resulting models have been integrated with concepts that are peculiar to specific types of cultural property.

In all cataloguing standards, metadata are grouped into paragraphs, each containing different fields. A manual clustering of such fields led to classify paragraphs and sub-paragraphs according to a set of topics. This activity was performed as a step towards identifying conceptual criteria that could guide the identification of the ontology modules that form the ArCo ontology network, i.e. to design its architecture. An early observation is that the content of a catalogue record both contains data directly describing a cultural property and its context (e.g. techniques and materials, related exhibitions, surveys) as well as data about the catalogue record itself (e.g. when it was created, by whom, its version, etc.) and about other entities related to the cultural property (e.g. inventories, documentation, bibliography). Hence a first distinction to be made

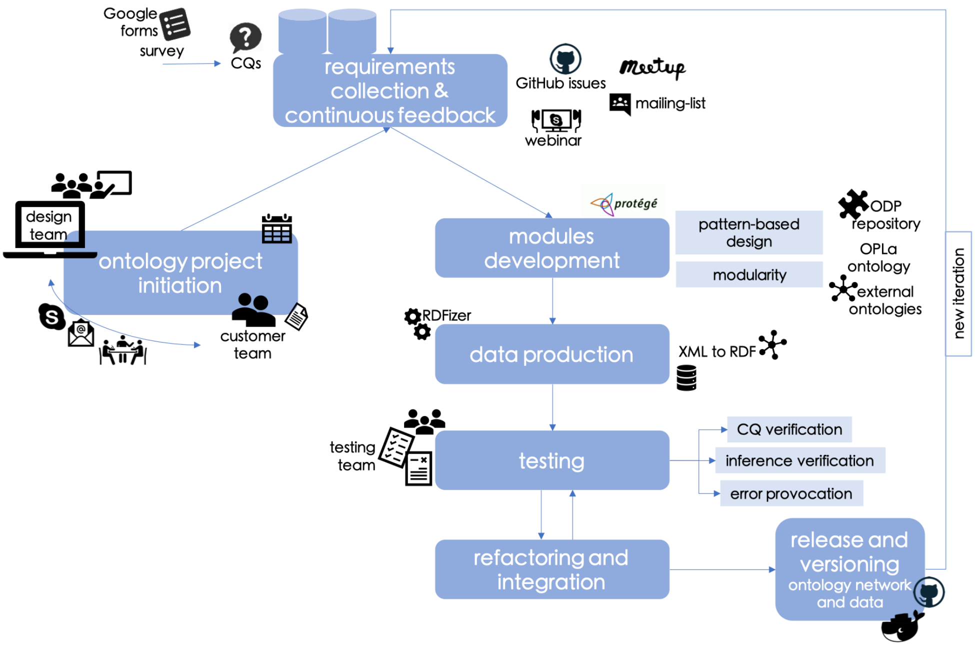


Fig. 5. The XD methodology as implemented for the ArCo knowledge graph.

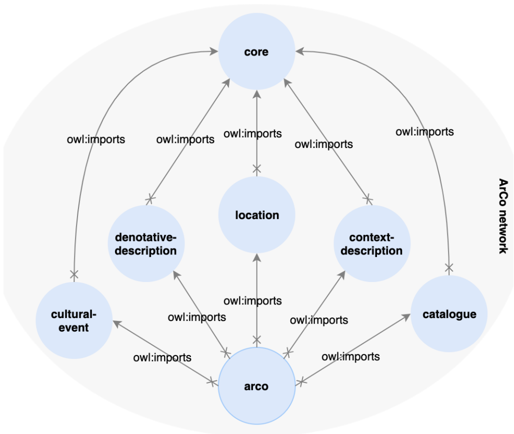


Fig. 6. ArCo ontology network: it currently includes seven modules. The module *arco* is the root node of the network and models top level distinctions and the general taxonomy of cultural properties, while *core* models general concepts that are reused by all other modules. The module *catalogue* models catalogue records and their evolving process, the module *denotative description* and *context description* encode objective and subjective descriptions of a cultural property, respectively. The modules *location* and *cultural events* model the different types of locations of a cultural property and the events it participates in, respectively.

within the ArCo ontology is between data about the catalogue and data about cultural properties. Catalogue records have their own ontological characteristics and relevance into the CH domain. They are *about* cultural properties, hence they encode their epistemic perspective. Based on this observation, a module (*catalogue*[28]) has been dedicated to the General Catalogue (GC), and in particular to catalogue records and its attributes. This module models also other records describing the cultural property, such as additional forms, which deepen the description of specific attributes of a cultural property (e.g. the archaeological stratigraphy).

Cultural properties, which are the main objects of study of the CH domain, are described both by means of measurable, intrinsic aspects such as length, weight, materials, and conservation status, and of subjective or external aspects such as authorship attribution, dating that may connect them to other entities and are usually based on an interpretation process and external sources. This conceptual distinction led to identifying two additional modules of the network: *denotative description*[29] as for capturing descriptions based on objective reference systems, and *context description*[30] that encodes situations and objects (e.g. inventories) which represent the context of the cultural property, in a broad sense, and can be used in order to retrace its life cycle.

[28] `a-cat`: https://w3id.org/arco/ontology/catalogue
[29] `a-dd`: https://w3id.org/arco/ontology/denotative-description
[30] `a-cd`: https://w3id.org/arco/ontology/context-description

The ICCD *core* standard makes it evident that the *locations* associated with a cultural property and the *cultural events* in which it participates in, are two major components of its lifecycle. For this reason ArCo ontology network includes a module *location*[31] dedicated to the different types of locations of a cultural property (e.g. current location, where it was found, where it was exhibited, where it was created, where it was stored, etc.), and representing physical sites (e.g. a palace), geometrical features and related cadastral entities such as a cadastral map. A different module *cultural event*[32] is dedicated to classes and properties representing attributes of cultural events, including events that recur over time (cf. Section 4.3.3), a type of events that we found neglected or underspecified in existing literature although it is of notable importance in the CH domain e.g. festivals, recurrent exhibitions, festivities. In addition, the *core*[33] module is dedicated to general concepts, e.g. part-whole relation, and is imported and specialised by all other modules, and the (*arco*[34]) module contains top-level distinctions, i.e. a taxonomy of different types of cultural properties, and imports all the other modules, thus playing as a root ontology and returning the whole ontology network.

As a result, ArCo ontology network is currently composed of seven modules, as depicted in Figure 6.

Competency Questions. Each module of the ontology network answers to a subset of the Competency Questions elicited by the customer team. Table 1 lists representative CQs for each module, except the *core* module, which is specialised by the other modules.

4.3. Matching requirements to Ontology Design Patterns

In this section, we describe the process that leads to match requirements to Ontology Design Patterns (ODPs) for some salient modelling issues addressed by ArCo.

Figure 7 shows all the prefixes used in the next diagrams.

Prefixes

: https://w3id.org/arco/ontology/arco/
a-cat: https://w3id.org/arco/ontology/catalogue/
a-loc: https://w3id.org/arco/ontology/location/
a-dd: https://w3id.org/arco/ontology/denotative-description/
a-cd: https://w3id.org/arco/ontology/context-description/
a-ce: https://w3id.org/arco/ontology/cultural-event/
core: https://w3id.org/arco/ontology/core/
l0: https://w3id.org/italia/onto/l0/
dul: http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#
tiapit: https://w3id.org/italia/onto/TI/
roapit: https://w3id.org/italia/onto/RO/
cis: http://dati.beniculturali.it/cis/
rdfs: http://www.w3.org/2000/01/rdf-schema#
rdf: http://www.w3.org/1999/02/22-rdf-syntax-ns#

Fig. 7. Prefixes used in the next figures.

4.3.1. Representing dynamics

Dynamic concepts, such as situations that change over time, are present in every domain. There are different patterns that model dynamic situations: in the remainder of this subsection, we present how we deal with catalogue records and cultural property locations, which may both evolve over time.

A catalogue record as a fluent information object. A catalogue record is an entity that describes a cultural property. As it denotes a real-word object, it can be defined as an *information object*, i.e. a piece of information, independent from how it is concretely realised, describing something in the real word. This concept is defined in several ODPs, including *Information Realization*[35] [25], which is reused in ArCo. The content of a catalogue record, i.e. the description of a cultural property, can change: "information about the creation of a catalogue record and possible following computerisation, update and corrections". Indeed, different agents with different roles (e.g. the official in charge) can be recorded, and a date keeps track of the time interval associated with each action.

A catalogue record is then a fluent entity, an information object that changes as the description of its denoted cultural property changes. Possible corrections and updates implemented by a cataloguer can derive from (i) an ontological change of the cultural property such as changes in some of its attributes (e.g. the conservation status from good becomes mediocre, since the cultural property has been badly preserved); (ii) an epistemological change, if the knowledge, which the catalogue record is based on, is either no longer complete, due to new knowledge acquired, or no longer valid, as a result of new research activities (e.g. after discovering new documentation, it turns out that another author played a role in creating the cultural property).

[31] `a-loc`: https://w3id.org/arco/ontology/location
[32] `a-ce`: https://w3id.org/arco/ontology/cultural-event
[33] `core`: https://w3id.org/arco/ontology/core
[34] `:` https://w3id.org/arco/ontology/arco

[35] http://www.ontologydesignpatterns.org/cp/owl/informationrealization.owl

Table 1

Representative competency questions answered by ArCo ontology network.

ID	Competency question	ID	Competency question
	ArCo module		**Cultural Event module**
CQ1	Is a cultural property tangible or intangible?	CQ18	In which cultural events and exhibitions a cultural property has been involved?
CQ2	Is a cultural property movable or immovable?	CQ19	Which cultural properties have been involved in a cultural event?
CQ3	Which are the cultural properties of a given type?	CQ20	Which are the events of a recurrent event series?
CQ4	Which are the components of a complex cultural property?	CQ21	Which is the time period elapsing between two events of a recurrent event series?
CQ5	Which is/are the residual(s) of a cultural property?	CQ22	Which are the unifying criteria shared by all the events in a recurrent event series?
	Location module		**Denotative description module**
CQ6	Which are all the places where a cultural property has been located? Which are their types?	CQ23	Which is the conservation status of a cultural property?
CQ7	When a cultural property has been located in a place?	CQ24	What is the technical status of a cultural property?
CQ8	Which are the geographical coordinates of a cultural property?	CQ25	Which are the measurements of a cultural property?
CQ9	Which are the cadastral data associated to the cultural property location?	CQ26	Which are the elements, e.g. inscriptions, affixed on a cultural property?
	Context description module		**Catalogue module**
CQ10	Which are the authors and/or cultural scopes attributed to a cultural property?	CQ27	Which is the level of detail of the catalogue record?
CQ11	When a cultural property has been created?	CQ28	When was a catalogue record created or updated?
CQ12	Which is the subject represented on a cultural property?	CQ29	Which are all the versions of a catalogue record?
CQ13	Which are the current and/or previous owners of a cultural property?	CQ30	Which is the (immediate) previous catalogue record version of a catalogue record version? And which is the (immediate) next one?
CQ14	Who commissioned a cultural property?	CQ31	Who, and playing which role, was responsible for creating, editing and updating a catalogue record?
CQ15	Which are the bibliography and documentation related to a cultural property?	CQ32	Which is the catalogue record describing a cultural property?
CQ16	Which interventions and surveys have been carried out on a cultural property?	CQ33	Which is, and for what reason, the level of privacy of a catalogue record?
CQ17	Which collection a cultural property is member of?		

Every change of a catalogue record produces an information object, which is a new version of the catalogue record. Nevertheless, the catalogue record finds its persistence in always denoting the same real-word object, i.e. the same cultural property, independently from different versions of the content and changes over time. Thus, the catalogue record denoting a cultural property is represented as a persistent information object, and is related to its versions, which are information objects reflecting changes of its content over time.

The *Time Interval* ODP is implemented to represent the temporal validity of each version, and the pattern *Sequence*[36] allows us to represent the sequence of these consecutive information objects.

In Figure 8 we depict how we model catalogue records by reusing the mentioned ODPs. A catalogue record is represented by the class `a-cat:CatalogueRecord`, which is aligned to `dul:InformationEntity`, with an `rdfs:subClassOf` axiom, and is related to the cultural property it describes. Catalogue records have different `a-cat:CatalogueRecordVersion`s. Each version is associated with a time interval `a-cat:editedAtTime`, and with agents

[36]http://ontologydesignpatterns.org/cp/owl/sequence.owl

involved in its creation (e.g. `a-cat:hasCataloguingAgent` with its subproperties). The agents involved in changing the catalogue record play some role which has its own temporal validity, hence we here see the implementation of another pattern. `roapit:TimeIndexedRole` is modelled as a time-indexed situation (see Section 4.3.2 for more details on situations) involving an agent, its role, and the temporal index of the agent-role relation. The object properties `a-cat:has(Immediate)PreviousVersion` and `a-cat:is(Immediate)PreviousVersionOf` specialise the *sequence* ODP allowing to represent (in)transitive previous and next versions of a catalogue record.

Multiple time-indexed and typed locations for one cultural property. A tangible cultural property, i.e. a physical object, is located in a physical place, which can be defined by a set of components: country, region, city, address, etc. For an immovable cultural property (e.g. a monumental park), this place overlaps with the area occupied by the cultural property, and to which it is fixed. Instead, for a movable cultural property (e.g. a photograph), information about the building in which it is situated and preserved, and the related cultural institute, is provided. While an immovable cultural property, precisely because of its nature, will be related to one and only one geographical place during its whole life cycle, a movable cultural property can be moved from a place to another. Different locations of a cultural property will hold in different time intervals. It is then clear that the temporal validity of the locations associated with a cultural property need to be represented, which also allow to reconstruct the spacial moving of the cultural property over time. Based on this, we can say that, during its life cycle, a movable cultural property is involved at least in as many situations as the places in which it has been located, and each situation is associated with a time interval. The *Time Indexed Situation*[37] ODP, [26] which represents situations that have an explicit time parameter, can be reused to this aim. Another important aspect is the motivation that links a cultural property to a location: it can be the place where it was found, created, of an exhibition it was involved in, where it was temporarily stored, etc. A same place can play different *roles* as location of one or different cultural properties, thus this *role* must be evident in the time indexed situation.

Figure 9 shows the class `a-loc:TimeIndexedTypedLocation` as the core class of the implementation of this pattern: it is a *situation* of a cultural property that is *located* in some place, at a certain point in *time*, and with the location playing a specific *role* in such situation, thus typing that situation. A time indexed typed location is therefore associated with a `a-loc:LocationType` (e.g. `a-loc:FindingLocation`, `a-loc:ExhibitionLocation`, etc.). `tiapit:atTime` relates the situation to its temporal validity, while `a-loc:atSite` and `a-loc:atLocation` express the site (intended as a physical building) and the geographical entity involved in the situation, respectively.

4.3.2. Situations and their descriptions

A cultural property can be involved in many different situations during its life: it can be commissioned, bought or obtained, used (e.g. a garment wore by one person), it can be part of a collection, photographic or numismatic series, can change its availability as a result of theft, destruction or rescue, etc. Each situation defines a contextual relation between the cultural property and the other entities involved. For example, when a coin is issued, many entities play a role in such context: the cultural property itself, the issuer, the issuing State, the mint and the minter. The "coin issuance" is a situation representing the relation that keeps together all these entities for that purpose. The ODP *Situation*[38] [26] models the concept of a contextual *n*-ary relation. Figure 10 shows how we model the *coin issuance* (`a-cd:CoinIssuance`) by implementing this ODP.

The technical status of a cultural property. Another example of situation involving a cultural property is its technical status. In this case a cultural property is related to a set of technical characteristics. For example, "the archaeological cultural property realised with pottery material and cylindrical in shape". These characteristics can change over time, thus modifying the technical status of the cultural property: for example, a new survey on an archaeological monument may discover new materials used for its foundation. The temporal validity of a technical status refers to the moment when the characteristics were observed (and recorded in the catalogue record) until when a new condition occurs.

Different technical characteristics of a cultural property can be specified, in order to describe its techni-

[37] http://www.ontologydesignpatterns.org/cp/owl/timeindexedsituation.owl

[38] http://www.ontologydesignpatterns.org/cp/owl/situation.owl

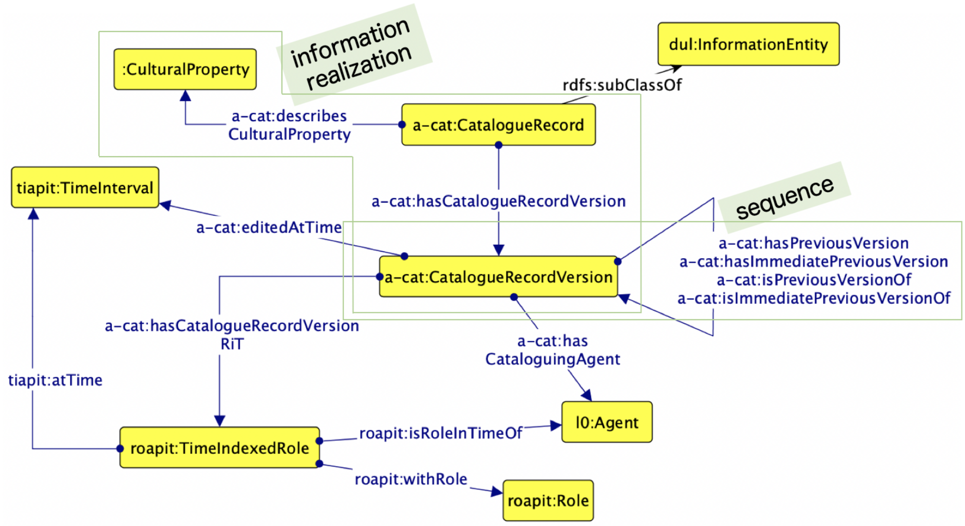


Fig. 8. Information Realization and Sequence ODPs reused for modeling catalogue records.

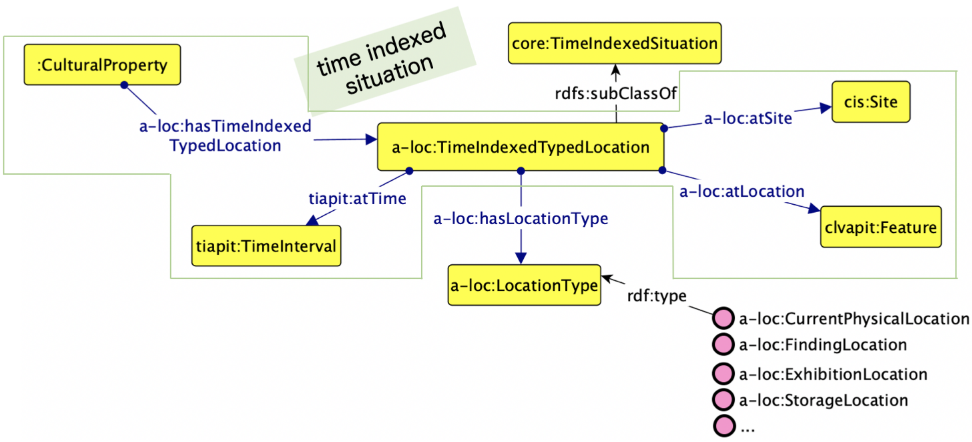


Fig. 9. Time indexed situation ODP implemented for modelling different types of locations of a cultural property.

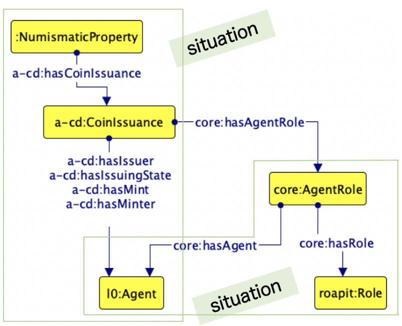


Fig. 10. Situation ODP reused for representing the coin issuance.

cal status: the constituting materials (e.g. wood, clay), the employed techniques (e.g. oil-painting, melting), the shape (e.g. square, octagon), the file format for a digital photograph (e.g. ".gif", ".jpeg"), the prevalent colour of a garment, etc. All these concepts (i.e. material, techniques. shape) *classify* the corresponding technical characteristics (i.e. wood, oil-painting, square). The *Classification*[39] ODP [27] defines a classification relation between a concept and an object, which exactly captures this circumstance.

A specific set of *technical concepts* classifying the *technical characteristics* of a cultural property type (e.g. an artwork) represents one way to conceptualise

[39] http://www.ontologydesignpatterns.org/cp/owl/classification.owl

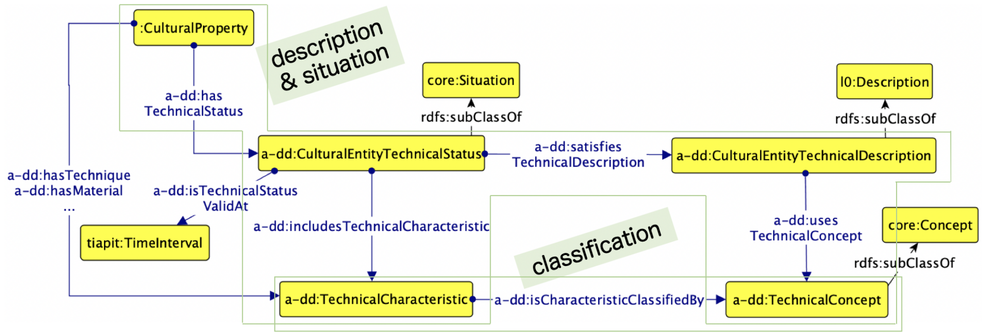


Fig. 11. The D&S pattern reused and specialised for modelling technical descriptions and status of a cultural entity.

the technical status of a cultural property, they constitute a *technical description*. For example, a *artwork technical description* may be defined as the relation between *constituting material*, *employed technique*, and *shape*. We say that such technical description *uses* these concepts. The *technical status* of artwork *A1* could be wood, oil-painting, and square, while the technical status for artwork *A2* could be clay, melting, and octagon. Both such technical status are expressed according to the *artwork technical description*, we say that they *satisfy* it. The *Description and Situation*[40] ODP [26] models the *satisfy* relation between situations and descriptions, and reuses the *Classification* ODP to model the relation between objects of a situation and concepts of the corresponding description.

Figure 11 shows how we model the `a-dd:CulturalEntityTechnicalStatus`, which includes the `a-dd:TechnicalCharacteristics` observed on a cultural property, as a subclass of `core:Situation`. Each characteristic is *classified by* a `a-dd:TechnicalConcept`, e.g. the `a-dd:Shape`. For the most common values of technical concepts we provide a controlled vocabulary. These concepts are used in the `a-dd:CulturalEntityTechnicalDescription`, defined as a subclass of `l0:Description`.

4.3.3. *Recurrence in cultural events and in intangible cultural heritage*

When modelling events, it is possible to distinguish between at least two event types: those occurring only once and those recurring over time. The first type refers to events that have a start date and an end date, and are not repeatable or reproducible; this is the case of the creation of a cultural property. The second type refers to series of events such as the *Art Biennale*[41], which have something in common and are separated by regular time intervals. Cultural properties can be involved in different cultural events, such as exhibitions, during their life cycle. Many of these events have different editions, which occur on a regular basis, e.g. an annual painting award. Moreover, cultural properties themselves can be recurrent: an intangible cultural heritage can have regular time intervals between its repetitive occurrences, such as a traditional ceremony related to the *year cycle* (e.g. Carnival).

As these particular events unfold, we can recognise a pattern in their iteration: an exhibition that has different editions over years usually follows a pattern in planning consecutive editions at regular time intervals (e.g. one edition per year). Moreover, it is possible to identify attributes that give all occurrences a unity: a general topic that does not change i.e. contemporary art, a place that host the event i.e. Venezia, etc.

Recurrent events are usually modelled as a special type of events (cf. Wikidata[42]), while their belonging to a series and the nature of such a collecting, unifying entity is neglected in literature or confused with the concept of event (cf. DBpedia resource for Venice Biennale[43]). We believe that modelling both the unitary series of events, e.g. the *Art Biennale*[41] intended as something that occurs biennially under certain conditions thus having different instances, and its individual events members, e.g. the *Art Biennale 2019* intended as a particular edition of the series with a start date and an end date, is important in the CH domain context.

[40] http://www.ontologydesignpatterns.org/cp/owl/descriptionandsituation.owl

[41] https://www.labiennale.org/en/art/

[42] https://www.wikidata.org/wiki/Q15275719

[43] http://dbpedia.org/page/Venice_Biennale

We hence create a new ODP for modelling *Recurrent event series*[44] [28], which reuses other existing ODPs.

We represent, as depicted in Figure 12, recurrent event series (`a-ce:RecurrentEvent`) as collections of events, their members (`a-ce:hasMemberEvent`). Event members share at least one common property (e.g. the topic) and are conceptually unified by a *unifying factor* (`a-ce:hasUnifyingFactor`) which characterises the series. At the same time, a recurrent event series is a situation, since it provides a relational context to all the event members. Each event member has its own time interval and is put in a *sequence* that relates it to the other event members of the same series (`a-ce:hasNextEvent`). The time period that elapses between event members is (approximately) regular and is an attribute of the series `a-ce:hasRecurrentTimePeriod`.

4.4. Direct and indirect reuse of patterns

Reusing ontologies and ODPs can be done by following two main different approaches, depending on the conditions and requirements of a project: direct and indirect reuse [29].

Direct reuse. This approach consists in directly embedding individual entities or importing implementations of ODPs or other ontologies in the local ontology, thus making it highly dependent on them. This may jeopardize the stability of the ontology if the evolution of the imported ontologies is outside the control or monitoring of the team/organisation that is reusing them: even small changes in the reused ontologies could introduce inconsistencies in the local one, contrary to its original requirements. For this reason, ArCo directly reuses only two ontologies that are considered reference standards by the Italian Government and the evolving process of which is relatively slow and systematised, and involves ArCo's team. These ontologies are Cultural-ON, which is also directly maintained by MiBAC, and OntoPiA ontology network, which is recommended as a standard for open data of the Italian Public Administration (and that now includes ArCo).

Indirect reuse. In this approach, relevant entities and patterns from external ontologies are used as *templates*, by reproducing them in the local ontology and providing possible extensions. Alignments axioms (such as `rdfs:subClassOf` and `owl:equivalentClass`) are introduced to support interoperability with other ontologies and make it evident what parts have been reused. This practice decreases the dependency on external ontologies, and is widely adopted in ArCo.

4.4.1. Annotating reused ODPs

All (re)used ODPs in ArCo are annotated with OPLa[45] [30], a simple ontology design pattern annotation language, which simplifies future reuse of ArCo by third parties as well as matching to other resources. For instance, it is possible to express that a pattern in a local ontology is a specialisation or a generalisation of a more general ODP. Let us consider the *catalogue* module (see Figure 13). Over this module, two ODPs from the *ODP portal*[46] have been (indirectly) reused: the module is therefore annotated with the property `opla:reusesPatternAsTemplate` for representing the reuse of *Sequence*[36] (see Figure 13a) and *Classification*[39] ODPs. For example, the pattern *Catalogue Record Sequence*[47] is a specialisation of the pattern *Sequence*, since it represents a sequence of catalogue records, hence it is annotated with the annotation property `opla:specializationOfPattern` (see Figure 13a). For expressing that specific properties (e.g. `a-cat:isPreviousVersionOf`, `a-cat:hasImmediatePreviousVersion`, etc.) implemented in the catalogue module, belong to this ODP, the annotation property `opla:isNativeTo` is used, as in Figure 13b.

4.5. Testing a Knowledge Graph

Testing an ontology by running a reasoner, in order to detect any incoherence in the model, is a necessary step, but not sufficient. We regularly run the HermiT reasoner[48], but we perform additional tests at each iteration of the methodology, i.e. every time new requirements are selected to be addressed. In doing so, we adopt the testing methodology described in [24], which focuses on evaluating the appropriateness of an ontology against its requirements intended as the ontological commitment expressed by means of CQs and domain constraints. In order to foster reproducibility and to offer a useful case study for other projects aiming at using this type of testing methodology, we publish all testing activities and resulting data, which are OWL

[44] http://www.ontologydesignpatterns.org/cp/owl/recurrenteventseries.owl

[45] `opla:` http://ontologydesignpatterns.org/opla/

[46] http://ontologydesignpatterns.org

[47] https://w3id.org/arco/pattern/catalogue-record-sequence/

[48] http://www.hermit-reasoner.com/

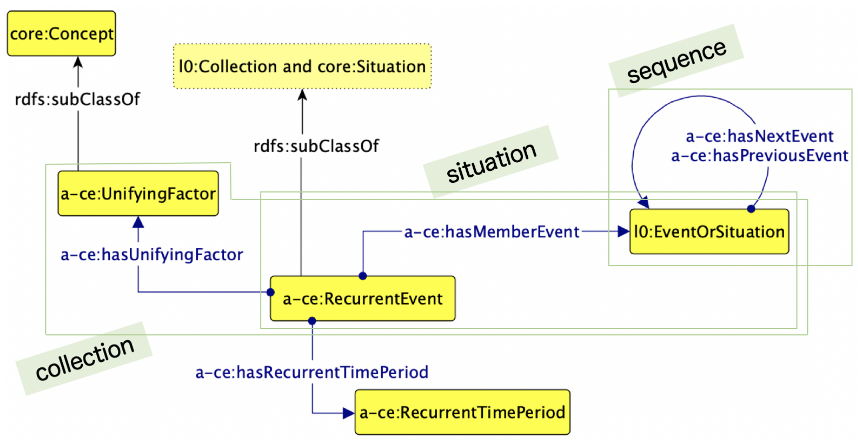
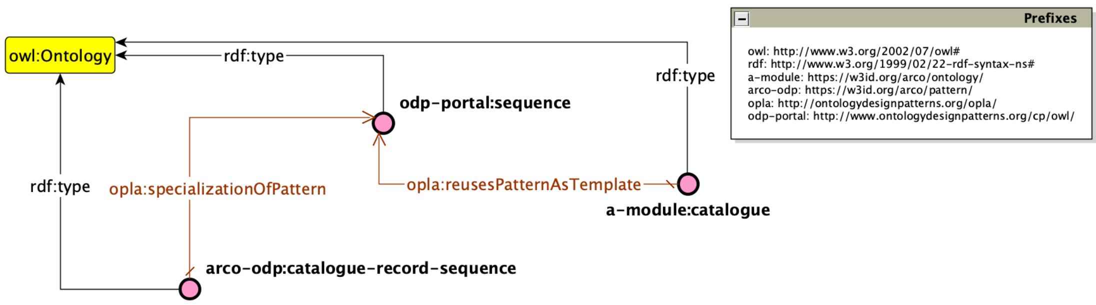
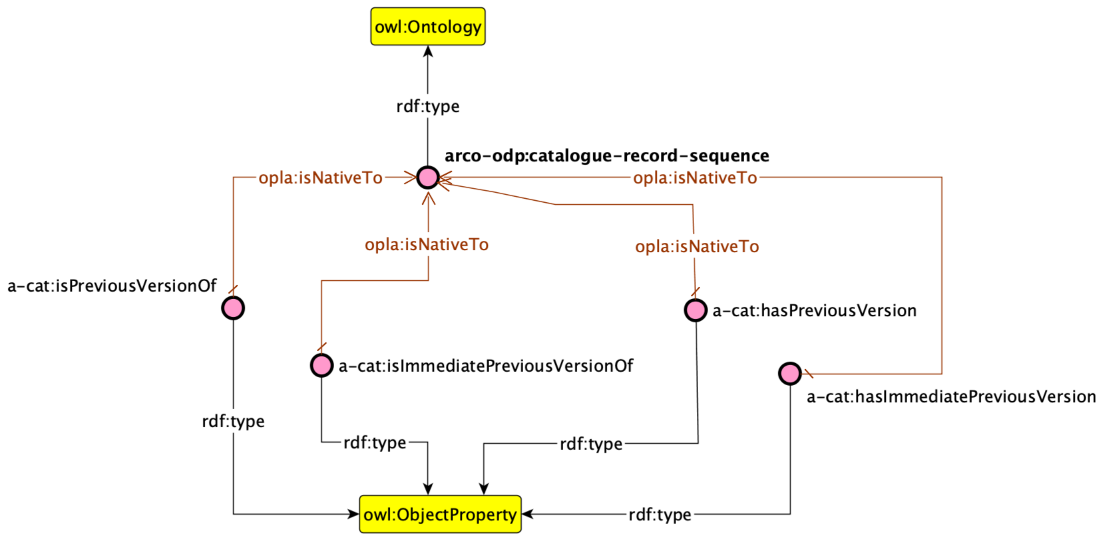


Fig. 12. The new pattern Recurrent Event Series as implemented in ArCo.



(a) The annotation property `opla:reusesPatternAsTemplate` relates the *catalogue* module to the Sequence ODP that is reused over the module. The annotation property `opla:specializationOfPattern` relates the pattern *catalogue record sequence* implemented in the module to the *Sequence* ODP that has been specialised.



(b) The annotation property `opla:isNativeTo` relates the object properties that belong to the *catalogue record sequence* ODP to the ODP itself.

Fig. 13. An example of a reused ODP annotated with OPLa ontology.

files complying to the ontology described in [24], on GitHub[49]. Three different approaches are followed in the testing activity: *CQ verification, inference verification, error provocation*.

CQ verification. This approach consists in testing whether the ontology vocabulary allows to convert a CQ, reflecting an ontology requirement, to a SPARQL query. Let us consider the CQ "When was a cultural property created, and which is the interpretation criterion which the dating is based on?", which ArCo ontology network should answer, based on the collected requirements. The testing team starts verifying the completeness of the ontologies by translating this question from natural language to SPARQL, using classes and properties defined in ArCo ontologies (e.g. the entities defined for representing the date of creation of a cultural property). This step allows to detect any missing concept or gap in the vocabulary, e.g. whether the concept of interpretation criterion has been modeled. If the CQ can be successfully converted, the testers run the resulting SPARQL query over the actual RDF data or, when missing, over test data generated using Fuseki[50], and complete the test by comparing the expected result (i.e. the output they expect from the query) to the actual result.

Inference verification. This step focuses on checking the inferences caused by the ontologies, by comparing the expected inferences to the actual ones. Let us consider a complex cultural property, which is a cultural property with one or more components, as proper parts. If a `:ComplexCulturalProperty` is defined as a `:CulturalProperty` that has one or more `:CulturalPropertyComponent`s, an axiom stating that a `:CulturalProperty` has a `:CulturalPropertyComponent` would suffice to infer that the property is complex, even if it is not explicitly asserted. For comparing this expected inference with the actual one, the testing team injects the necessary data in the knowledge graph – i.e. an instance of the class `:CulturalProperty` related to an instance of the class `:CulturalPropertyComponent` through the object property `:hasCulturalPropertyComponent` – and runs the reasoner. If the reasoner does not infer that the first instance is `rdf:type :ComplexCulturalProperty`, this means that the appropriate axiom is missing from the ontology, i.e. an equivalent axiom between `:ComplexCulturalProperty` and (`:hasCulturalPropertyComponent` some `:CulturalPropertyComponent`).

Error provocation. This third testing activity is intended to "stress" the knowledge graph by injecting inconsistent data that violate our requirements.

For instance, the entities representing the concepts of dating and attributing an author to a cultural property should be disjoint, since there can be no individuals that are dating and authorship attributions at the same time. For validating the ontology regarding this requirement, the testers inject in the KG an individual belonging to both `a-cd:AuthorshipAttribution` and `a-cd:Dating` classes, and runs the reasoner. The expected result is an inconsistency: if this is not detected by the reasoner, it means that the appropriate (disjointness) axiom is missing.

Refactoring and integration. Problems spotted during the testing phase are passed back to the design team as issues. The design team refactors the modules and updates the ontology after performing a consistency checking. The result of this step is validated again by the testing team before including the model in the next release.

4.5.1. TESTaLOD

In the context of ArCo project, performing the testing activities initially resulted in a significant manual effort, for both annotating and running the unit tests. For this reason, TESTaLOD, a tool for supporting not only the testing team of ArCo project, but in general any testing team of projects adopting XD methodology or other test-driven methodologies, has been designed and implemented.

TESTaLOD is developed as a Web application[51] that provides a knowledge graph testing toolbox: it implements a two-step workflow, allowing the user to select and automatically run defined test cases aiming at verifying CQs. The test cases are OWL files, and are modelled by using the TestCase OWL meta model introduced in [10], thus containing: a Competency Question and its corresponding SPARQL query, the expected (correct) result and data sample. The test cases can be either retrieved from a GitHub repository or uploaded from a local file system. Once the tests have been automatically executed, the expected result is compared to the actual result, and three possible outputs can be

[49] https://github.com/ICCD-MiBACT/ArCo/tree/master/ArCo-release/test

[50] https://jena.apache.org/documentation/fuseki2/

[51] https://w3id.org/testalod

displayed to the user: successful, partially successful, unsuccessful.

Let us consider as an example the competency question "Which archival set (fonds, series, subseries) a cultural property is member of?", and that we want to verify if our ontology models information on membership of cultural properties to archival record sets. The test case for running this test will be an OWL file, annotated with the following properties[52]: `test:hasCQ` has the competency question expressed in natural language as a value; `test:hasSPARQLQueryUnitTest` the translation of the CQ to SPARQL, using the ontology entities; `test:hasInputTestData` points out the test data used as input for running the test; `test:hasExpectedResult` stores a set of expected results of running the query over a certain set of test data; `test:hasActualResult` stores the actual outcome of a test run. Other properties are used in order to annotate who run the test and when.

In order to allow TESTaLOD to automatically run this test, two new annotation properties[53] have been defined. `testalod:hasInputTestDataCategory` annotates if the input data are available at a SPARQL endpoint (`testalod:SPARQLendpoint`) or in a file with test data (`testalod:ToyDataset`); `testalod:hasInputTestDataUri` annotates the URI of the SPARQL endpoint or the file, which is used by TESTaLOD to run the query.

4.6. *Involving external actors for building a KG on cultural heritage*

As mentioned above, the requirements collection is a crucial step in eXtreme Design projects. These requirements guide the ontology development: once collected, the design team selects and satisfies them sequentially in the ontology. Collecting the requirements is performed at each iteration of the ontology design process; indeed, as ArCo is published with incremental releases, new emerging requirements can be elicited by the customer team, and feedback on previous versions can lead to changes and updates in the model, or error correction. Improvement proposals and bugs are submitted as issues through GitHub[54].

ArCo's requirements are collected in the form of small stories, as scenarios and real use cases (according to XD). They are then translated as Competency Questions[55] (CQs), i.e. questions to be answered by the ontology [9], and used for selecting ODPs by the design team, as well as in the testing phase by the testing team.

Our main *customer* was ICCD, i.e. the institute in charge of collecting and preserving the data of the General Catalogue, and of releasing updated cataloguing standards. The domain experts of ICCD guided the design team in selecting and prioritising relevant requirements based on their data and their expertise. At each iteration, domain experts were invited to give feedback on the ontologies addressing the selected requirements.

In order not to limit the ontology design only to institutional and regulatory requirements, the customer team was extended by involving a wider community, potentially interested in reusing ontologies and data on Cultural Heritage (CH), such as private companies and public administrations working with CH data and services. For this purpose, after the project initiation and before the first release, a first meetup was held, followed by a series of regular meetups and webinars. This growing community, involving interested stakeholders and consumers, interacts also *via* a dedicated mailing-list[56].

Stories are submitted by the customer team to a Google Form[57]. In this form, the users are asked to express the use they will make of ArCo: publishing their data based on, and complying with, ArCo's ontologies, linking their LOD to ArCo's, reusing ArCo's data in some applications to provide services. Then, they can choose a name for their project, and send one or more stories, possibly uploading additional files (e.g. a sample of their data in the original format). For instance, one of the stories collected by means of the Google form is the following:

> *Type: Linking my data to ArCo data*
>
> *Title: Cultural heritage and residential property*
>
> *Story: I am looking for a residential property to buy, and I want to filter the results based on the type of cultural heritage nearby.*

---

[52] `test:` http://www.ontologydesignpatterns.org/schemas/testannotationschema.owl#

[53] `testalod:` https://raw.githubusercontent.com/TESTaLOD/TESTaLOD/master/ontology/testalod.owl#

[54] https://github.com/ICCD-MiBACT/ArCo/issues

[55] https://github.com/ICCD-MiBACT/ArCo/blob/master/ArCo-release/test/CQ/CQs-SPARQLqueries.txt

[56] https://groups.google.com/forum/#!forum/arco-project

[57] https://goo.gl/forms/zCixt3B1ABYbj9JS2

In order to answer the related CQ, spatial and geometrical information about cultural properties should be modeled, thus becoming one of the requirements to be satisfied.

An "Early Adoption Program" has been launched since ArCo's early phase of development. It aimed at ensuring continuous interaction and proper engagement of at least some of the *external* consumers involved in the customer team. An *official* "Early Adopter" (EA) would commit to participate in ArCo meetups, and to test and provide feedback for every (unstable) release. In exchange, EAs are given assistance and support for reusing ArCo in their projects, and the fulfillment of their issues and requirements are put high in priority. EAs may change and/or increase in number, over time (by means of new open calls). Candidate EAs submitted a short project description involving ArCo, and four organisations were finally selected. The rationale behind the selection was to guarantee a reasonable, but manageable, representativeness of the different uses that organisations and consumers can make of ArCo.

The "Early Adopter" program has been very important to experiment and stabilise the use of Github issues and the mailing list to systematise the open gathering of community requirements. In fact, ArCo is still an evolving creature and such means are kept in place to support this activity. In the future, we plan to establish a process for accepting external contribution to the ontology network, hence further opening the development and evolution of ArCo to the community.

5. ArCo by examples

In the following sections, we provide examples of usage of the ontology network ArCo for exporting in RDF the data stored in the ICCD General Catalogue, with particular focus on some of the Ontology Design Patterns reused, as explained in Section 4.3.

ArCo's Knowledge Graph (ontology network and LOD data) is available through MiBAC's official SPARQL endpoint[58]. Moreover, ArCo KG is released as part of a package, which consists of a *docker* container available on GitHub[59], and its running instance online[60] - both English and Italian versions. This package includes: documentation, user guides and diagrams; the source code and a human-readable HTML documentation of the ontologies[61]; a SPARQL endpoint; examples of Competency Questions and their corresponding SPARQL queries; RDFizer[62], a software for converting XML data represented according to ICCD cataloguing standards[6] to RDF compliant to ArCo ontologies.

5.1. A cultural property at a glance

As depicted in Figure 14, the concept of `:CulturalProperty` is modelled as a *partition* of two classes: `:TangibleCulturalProperty` and `:IntangibleCulturalProperty`. A tangible cultural property is a physical object, i.e. material in nature (e.g. a photograph, an amphitheater, ancient garments). An intangible cultural property is defined as an ephemeral performance (e.g. a traditional dance, oral literature, a musical, choral or theatrical performance, handcrafted techniques, etc.), which is documented by audio and/or video recording. While intangible cultural heritage is a "living" entity, tangible cultural heritage is distinguished by its steadiness, due to its material nature.

`:TangibleCulturalProperty` is further specialized in `:MovableCulturalProperty`, i.e. objects that can be handled and moved by nature (e.g. a painting, a musical instrument, etc.), and `:ImmovableCulturalProperty`, i.e. objects fixed or incorporated into the ground, which generally occupy a large area (e.g. an archaeological site, a palace and its gardens, etc.).

Additionally, more specific types are defined down the hierarchy:

- `:DemoEthnoAnthropologicalHeritage`, which can be either intangible (a poem orally transmitted) or tangible (a body adornment);
- `:ArchaeologicalProperty`, which is tangible, either movable (a Punic mask) or immovable (a Roman aqueduct);
- `:ArchitecturalOrLandscapeHeritage` (a bell tower);
- `:HistoricOrArtisticProperty` (a sculpture group);
- `:MusicHeritage` (a pipe organ);

[58] http://dati.beniculturali.it/sparql
[59] https://github.com/ICCD-MiBACT/ArCo
[60] https://w3id.org/arco
[61] This documentation has been created with LODE: http://www.essepuntato.it/lode
[62] https://github.com/ICCD-MiBACT/ArCo/tree/master/ArCo-release/rdfizer

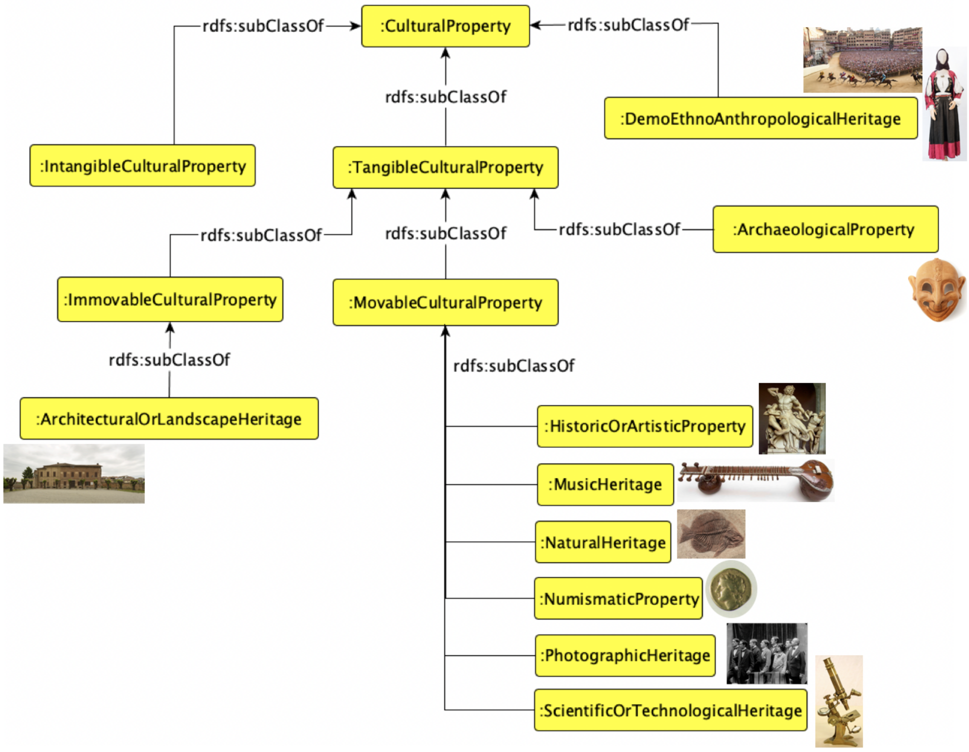


Fig. 14. The taxonomy of cultural properties.

- `:NaturalHeritage` (a fossil);
- `:NumismaticProperty` (a bronze sestertius);
- `:PhotographicHeritage` (a negative);
- `:ScientificOrTechnologicalHeritage` (a microscope).

By running the following SPARQL query[63] against ArCo KG, it is possible to find all cultural properties.

```
SELECT ?culturalProperty
WHERE
  {?culturalProperty
   rdf:type/rdfs:subClassOf*
    arco:CulturalProperty}
```

Let us take as an example a painting by Albert Friscia[64]. Listing 1 provides the definition of the resource as a `:MovableCulturalProperty`, in particular a `:HistoricOrArtisticProperty`. The RDF schema property `rdfs:label` is specified with a concatenation of values: the title of the painting ("Self portrait"), its subject ("Autoritratto"), the cultural property type ("dipinto", i.e. painting), the author ("Albert Friscia") and the century in which it was created ("sec. XX"). A cultural property has some identifiers associated with it (e.g. `:uniqueIdentifier`).

Listing 1: The code snippet for the definition of a *Cultural Property*.

```
data:HistoricOrArtisticProperty
    ↪ /1700168615
 a arco:HistoricOrArtisticProperty,
 arco:MovableCulturalProperty;
 rdfs:label "Self portrait, Autoritratto
     ↪  (dipinto) di Albert Friscia (sec
     ↪ . XX)"@it;
 arco:uniqueIdentifier "1700168615";
 ...
```

[63] In all the following SPARQL queries and listings, the prefix `data:` is used for https://w3id.org/arco/resource/; the prefix `arco:` for https://w3id.org/arco/ontology/arco/; the prefix `a-cat:` for https://w3id.org/arco/ontology/catalogue/; the prefix `a-loc:` for https://w3id.org/arco/ontology/location/; the prefix `a-dd:` for https://w3id.org/arco/ontology/denotative-description/; the prefix `a-cd:` for https://w3id.org/arco/ontology/context-description/; the prefix `ti:` for https://w3id.org/italia/onto/TI/.

[64] https://w3id.org/arco/resource/HistoricOrArtisticProperty/1700168615

5.2. A catalogue record and its versions

Different types of `a-cat:CatalogueRecord` are defined, based on the typology of cultural property they describe. The following query returns the catalogue record describing a cultural property, and all its versions.

```
SELECT ?catalogueRecord
        ?catalogueRecordVersion
WHERE
  {?catalogueRecord
   a-cat:describesCulturalProperty
    ?culturalProperty;
   a-cat:hasCatalogueRecordVersion
    ?catalogueRecordVersion .}
```

For example, the catalogue record in Listing 2 is of type `a-cat:CatalogueRecordPST`, i.e. a type of catalogue record describing scientific and technological heritage: indeed, this resource describes the cultural property `data:ScientificOrTechnologicalHeritage/0301971676`, i.e. a *Pensky-Martens tester*. It has 3 versions: a first version is created when the cultural property is first catalogued, and other versions follow, as a result of editing and updating activities. Let us consider the first version, i.e. `data:CatalogueRecordVersion/0301971676-compilation`: it has been encoded in 2002 and involved agents playing different roles in its compilation (e.g. the responsible of research and compilation). This relation is expressed as both a binary relation and a n-ary relation, whose range is a `ro:TimeIndexedRole`[65], i.e. a situation involving an agent, its role and the time of its duration.

Listing 2: The code snippet for the definition of a *Catalogue Record* and its versions.

```
data:CatalogueRecordPST/0301971676
 a a-cat:CatalogueRecordPST;
 rdfs:label "Catalogue Record n:
    ↪ 0301971676"@en;
 a-cat:describesCulturalProperty
  data:ScientificOrTechnologicalHeritage
    ↪ /0301971676;
 a-cat:hasCatalogueRecordVersion
  data:CatalogueRecordVersion
    ↪ /0301971676-compilation,
  data:CatalogueRecordVersion
    ↪ /0301971676-agg-1,
  data:CatalogueRecordVersion
    ↪ /0301971676-rvm .

data:CatalogueRecordVersion/0301971676-
    ↪ compilation
 a-cat:editedAtTime data:TimeInterval
    ↪ /2002;
 a-cat:hasCatalogueRecordVersionRiT
  data:TimeIndexedRole/0301971676-
    ↪ compilation-81909
    ↪ ca6a118f792436818a9170374a8;
 a-cat:
    ↪ hasResponsibleResearchAndCompilation
    ↪
  data:Agent/81909
    ↪ ca6a118f792436818a9170374a8;
 ...
```

5.3. Current and alternative locations for one cultural property

With the following SPARQL query, we can find all the locations associated with a cultural property during its life cycle, with their type and time interval.

```
SELECT ?culturalProperty
        ?location
        ?locationType
        ?locationTime
WHERE
  {?culturalProperty
   a-loc:hasTimeIndexedTypedLocation
    ?location .
   ?location
    a-loc:hasLocationType
     ?locationType ;
    ti:atTime
     ?locationTime .}
```

Let us consider a portrait by the Dutch artist Netscher Caspar[66]. In Listing 3 we report three different situations in which this portrait has been located, at different times, in different places: the place where it was located at the time of cataloguing, the place where it has been involved in an exhibition, the place where it was temporarily stored. In particular, `data:TimeIndexedTypedLocation/0900131533-alternative-1` is defined as a `a-loc:TimeIndexedTypedLocation`, which has `a-loc:ExhibitionLocation` as location type. The associated time interval

[65] `ro:` https://w3id.org/italia/onto/RO/

[66] https://w3id.org/arco/resource/HistoricOrArtisticProperty/0900131533

data:TimeInterval/1773 allows us to assert that this painting has been involved in an exhibition during 1773. The cis:Site (intended as the physical building containing movable objects) where the painting was exhibited is *Palazzo Pitti*.

Listing 3: The code snippet for the definition of *Time Indexed Typed Locations* of a cultural property.

```
data:HistoricOrArtisticProperty
    ↪ /0900131533
 a-loc:hasTimeIndexedTypedLocation
  data:TimeIndexedTypedLocation
      ↪ /0900131533-current,
  data:TimeIndexedTypedLocation
      ↪ /0900131533-alternative-1,
  data:TimeIndexedTypedLocation
      ↪ /0900131533-alternative-4,
  ...

data:TimeIndexedTypedLocation
    ↪ /0900131533-alternative-1
 a a-loc:TimeIndexedTypedLocation ;
 rdfs:label "Alternative location 1 of
     ↪ cultural property: 0900131533"@en
     ↪ ;
 a-loc:hasLocationType a-loc:
     ↪ ExhibitionLocation ;
 ti:atTime data:TimeInterval/1773
 a-loc:atSite data:Site/4
     ↪ baeab4f1066985b617b2d01e4e297fc .

data:Site/4
    ↪ baeab4f1066985b617b2d01e4e297fc a
    ↪  cis:Site ;
 rdfs:label "Palazzo Pitti" ;
 ...
```

5.4. Situations

As an example of possible situations involving a cultural property, we want to focus on a particular type of a-cd:Interpretations, i.e. situations in which pieces of information about an object (e.g. a cultural property) are detected and recorded by an agent, based on a specific source or motivation.

An a-cd:AuthorshipAttribution is a situation in which one author is attributed to a cultural property, and this attribution is motivated by an a-cd:InterpretationCriterion, e.g. inscription, bibliography, documentation. Each cultural property has at least one a-cd:PreferredAuthorshipAttribution and/or a a-cd:CulturalScopeAttribution, and can have one or more a-cd:AlternativeAuthorshipAttributions, previous and obsolete. The following SPARQL query returns the authorship attributions of a cultural property.

```
SELECT ?culturalProperty
        ?authorshipAttribution
WHERE
  {?culturalProperty
   a-cd:hasAuthorshipAttribution
    ?authorshipAttribution .}
```

In Listing 4, the cultural property data:NumismaticProperty/1500626679, i.e. a medal of the 17th century, has two preferred authorship attributions, since it has two authors (the object property a-cd:hasAuthor works as a shortcut for relating the cultural property to the preferred authors). The data:PreferredAuthorshipAttribution/1500626679-1 has Bonzagni Giovan Federigo as attributed author, which has been attributed based on data:InterpretationCriterion/analisi-stilistica, i.e. "stylistic analysis". Instead, the bibliography (data:InterpretationCriterion/bibliografia) has determined the attribution of Cesati Alessandro as an author. Both authors had "implementation" (data:Role/esecuzione) as role played in the creation of the cultural property.

Listing 4: The code snippet for the definition of *Authorship Attributions* of a cultural property.

```
data:NumismaticProperty/1500626679
 a-cd:hasAuthorshipAttribution
  data:PreferredAuthorshipAttribution
      ↪ /1500626679-1,
  data:PreferredAuthorshipAttribution
      ↪ /1500626679-2 ;
 a-cd:hasAuthor
  data:Agent/5
      ↪ fd1154534ebf5dfb384a59b3c1dd583,
  data:Agent/95
      ↪ f6d55ba0304ffa58d7698bbb83a5eb .

data:PreferredAuthorshipAttribution
    ↪ /1500626679-1
 a a-cd:PreferredAuthorshipAttribution ;
 a-cd:hasAttributedAuthor
  data:Agent/5
      ↪ fd1154534ebf5dfb384a59b3c1dd583
      ↪ ;
 a-cd:hasInterpretationCriterion
```

```
 data:InterpretationCriterion/analisi-
    ↪ stilistica ;
 a-cd:hasInterventionRole data:Role/
    ↪ esecuzione .

data:PreferredAuthorshipAttribution
    ↪ /1500626679-2
 a a-cd:PreferredAuthorshipAttribution ;
 a-cd:hasAttributedAuthor
  data:Agent/95
    ↪ f6d55ba0304ffa58d7698bbb83a5eb ;
 a-cd:hasInterpretationCriterion
  data:InterpretationCriterion/
    ↪ bibliografia ;
 a-cd:hasInterventionRole data:Role/
    ↪ esecuzione .
```

5.5. *Cultural entity technical status*

In order to have an overview on the technical status of a cultural property, i.e. of all its technical characteristics defining it, we can run the following query.

```
SELECT ?culturalProperty
        ?technicalStatus
        ?technicalCharacteristic
        ?technicalConcept
WHERE
  {?culturalProperty
   a-dd:hasTechnicalStatus
    ?technicalStatus .
   ?technicalStatus
    a-dd:includesTechnicalCharacteristic
     ?technicalCharacteristic .
   ?technicalCharacteristic
    a-dd:isCharacteristicClassifiedBy
     ?technicalConcept .}
```

The technical status in Listing 5 represents the situation in which a compass by an Italian workshop of the 19th century[67] has a technical status. This status includes two technical characteristics: “ottone” (brass) and “tondo” (circular). The first one is classified by the `a-dd:TechnicalConcept` material, while the second one by the shape. Thus, through the technical status we know that this cultural property is made of brass and is circular.

Listing 5: The code snippet for the definition of the *Technical Status* of a cultural property.

```
data:CulturalEntityTechnicalStatus
    ↪ /0300115504
 a a-dd:CulturalEntityTechnicalStatus ;
 rdfs:label "Technical status of
    ↪ cultural property 0300115504"@en
    ↪ ;
 a-dd:includesTechnicalCharacteristic
  data:TechnicalCharacteristic/ottone,
    ↪ data:TechnicalCharacteristic/
    ↪ tondo .

data:TechnicalCharacteristic/ottone
 a-dd:isCharacteristicClassifiedBy
  a-dd:Material .

data:TechnicalCharacteristic/tondo
 a-dd:isCharacteristicClassifiedBy
  a-dd:Shape .
```

6. A formal evaluation of ArCo

ArCo is evaluated along different dimensions: structural, logical, and the functional dimensions as identified by [13]. The structural dimension of a knowledge graph (KG)[68] focuses on its topological properties measured by means of context-free metrics that leverage its graph-based representation. The logical dimension measures whether an ontology can be successfully processed by a reasoner (inference engine, classifier, etc.). Finally, the functional dimension is related to the intended use of a given ontology and of its components, i.e. their function in a context.

6.1. *Structural dimension*

Experimental setup and results. For assessing the structural dimension of ArCo we use different metrics that have been defined and used in literature [11–17]. First, we compute base metrics that record quantitative aspects of ArCo knowledge graph: classes and their instances, properties, axioms, etc. Then, we compute schema and graph metrics aimed at assessing (i) the richness, width, depth, and inheritance at the schema level and (ii) the cohesion, coupling, multihierarchical degree, and extensional coverage of the ontolo-

[67] https://w3id.org/arco/resource/HistoricOrArtisticProperty/0300115504

[68] The authors of [13] refer to ontologies in their analysis. In the scope of this paper we generalise their results to knowledge graphs, since we also compute the distribution of the instances across classes.

gies. Those parameters are used for understanding the quality of ArCo expressed in terms of (i) flexibility, (ii) transparency, (iii) cognitive ergonomics, and (iv) compliance to expertise. These quality properties have been defined in [13]: (i) flexibility is the property of an ontology to be easily adapted to multiple views; (ii) transparency is the property of an ontology to be analysed in detail, with a rich formalisation of conceptual choices and motivation; (iii) cognitive ergonomics is the property of an ontology to be easily understood, manipulated, and exploited by its consumers; and (iv) compliance to expertise is the property of an ontology to be compliant with the knowledge it is supposed to model.

Table 2 and Table 3 describe the metrics used along with their corresponding results. Table 3 also reports the quality properties that the metrics are an indicator of. We use the association between metrics and quality property defined by [13]. The metrics are computed by using OntoMetrics[69], a web-based tool aimed at computing statistics about an ontology.

Then, we record the statistics about the distribution of the instances across classes. This allows us to understand how individuals are organised in the knowledge graph with respect to concepts. This suggests compliance to expertise. In fact, it provides a structural indication about the recall of classes over the entities of the domain (i.e. the individuals). In this case the recall is meant as extensional coverage computed as the average number of entities captured by ontology classes. It is worth saying that compliance to expertise has a strong functional characterisation that we investigate further by analysing the functional dimension. Notwithstanding, the distribution of the instances across classes is a fair structural metric as it provides us a tool for empirically validating if dense areas (most populated part of the ontology) correspond to ontology design patterns. The use of patterns is among the indicators suggested by [13] for measuring the quality properties of transparency and cognitive ergonomics. Figure 15 shows the top-50 ranked classes based on the number of individuals they have in the knowledge graph. The ranking including all the classes can be retrieved by querying the knowledge graph[70].

A high degree of modularity in an ontology is an indicator of transparency and flexibility. ArCo is highly modularised, however addressing transparency and flexibility meaningfully requires appropriate design of ontology modules. We compute the following metrics to assess the quality of ArCo modules.

- *Atomic size*: the average size of a group of interdependent axioms in a module;
- *Appropriateness of module size*: computed with the Schlicht and Stuckenschmidt function [15] that determines the appropriateness of an ontology module. The appropriateness value ranges from 0 (i.e. no appropriateness) to 1 (i.e. fully appropriateness). According to the Schlicht and Stuckenschmidt function a module size is as much more appropriate as the number of axioms defined in such a module is close to 250;
- *Encapsulation*: the measure of knowledge preservation within the given module computed as defined by [17]. Encapsulation values range from 0 (poor encapsulation) to 1 (good encapsulation);
- *Coupling*: the measure of the degree of interdependence of a module computed as proposed by [17]. Possible values range from 0 (high interdependence) to 1 (low interdependence).

Table 4 reports the values recorded for the aforementioned module metrics computed for each ontology module of ArCo. Module metrics are obtained by using the Tool for Ontology Module Metrics[71] (TOMM) [17].

Discussion. The analysis of the structural dimension shows that ArCo is a large knowledge graph both at schema (i.e. 3,416 logical axioms, 340 classes, etc.) and instance level (i.e. 20,030,941 individuals resulting in a distribution of 58,914.53 individuals per class on average). For more clarity we remind that the number of classes is used interpreting many of the metrics whose values are on ordinal scale (cf. Table 3). The indicators recorded suggest good transparency. In fact, we record:

- a sufficient axiomatisation of classes having 39.55 axioms per class (i.e. axiom/class ratio);
- a low inheritance richness (=2.48) that suggests a deep (or vertical) ontology, which, in turns, indicates that the ontology covers a specific domain in a detailed manner;

[69]https://ontometrics.informatik.uni-rostock.de/ontologymetrics/index.jsp

[70]The result set with the ranking of all classes is available at https://bit.ly/2ORiqnM.

[71]The specific version of the tool we used can be downloaded from https://bit.ly/2nSS2yD.

Table 2

Base knowledge graph metrics.

Metric	Description	Value
# of axioms	The total number of axioms defined for classes, properties, datatype definitions, assertions and annotations.	13,792
# of logical axioms	The axioms which affect the logical meaning the ontology network.	3,416
# of classes	The total number of classes defined in the ontology network.	340
# of object properties	The total number of object properties defined in the ontology network.	616
# of datatype properties	The total number of datatype properties defined in the ontology network.	154
# of individuals	The total number of individuals instantiated in the knowledge graph.	20,030,941
# of annotations	The total number of annotations in the ontology network.	8,734
DL expressivity	The description logics expressivity of the ontology network.	SROIQ(D)

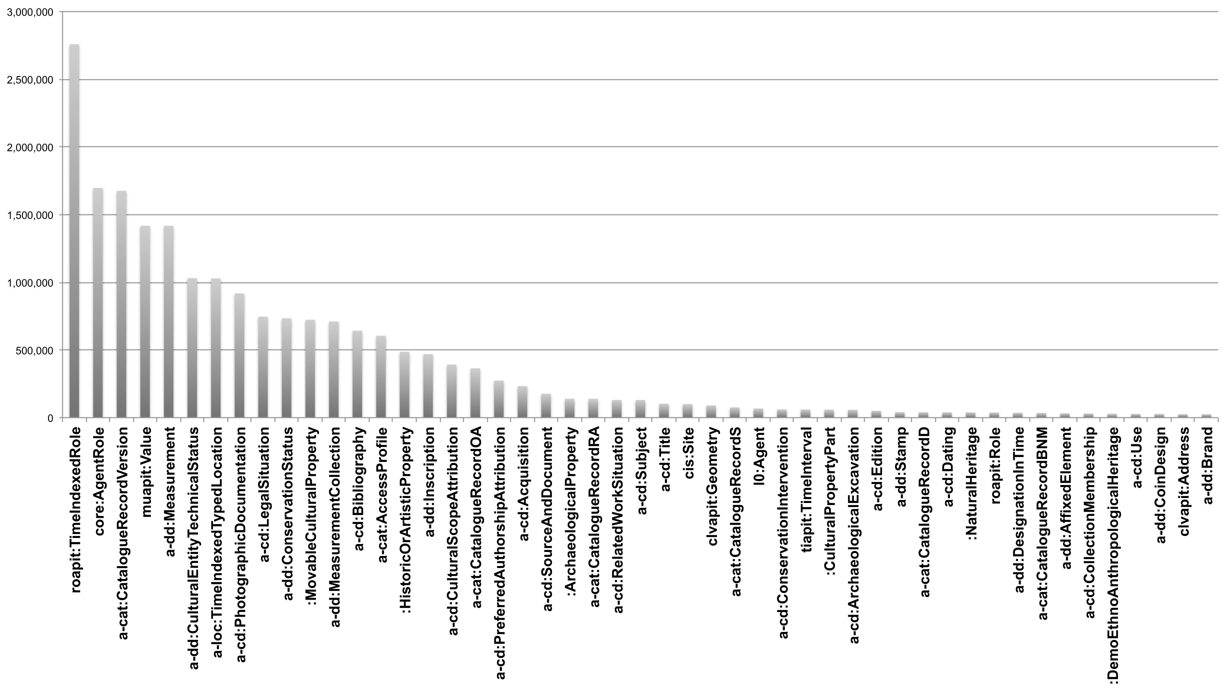


Fig. 15. Top-50 ranked classes according to the number of individuals they have in the knowledge graph.

- a positive indicator of the diversity of the relations defined in the ontology besides inheritance (i.e. relationship richness =0.44);
- low coupling suggested by the low number of external classes (NoC) used (i.e. 38 external classes on a total of 340 classes);
- a high degree of relatedness among the different classes, i.e. strong cohesion. In fact, the classes are organised in a hierarchy with (i) a low depth (i.e. ADIT-LN=3.93), (ii) a limited number of root classes if compared to the total number of classes (i.e. NoR=16), and (iii) a high number of to leaf classes if compared to the total number of classes (i.e. NoL=277).

Low coupling (i.e. NoC) and high cohesion (NoR, NoL, and ADIT.LN) also suggest flexibility, i.e. the property of adapting or changing the ontology with limited side-effects. The property of cognitive ergonomics (i.e. property of a knowledge graph to be easily understood, manipulated, and exploited by final users) is suggested by:

- the moderate class/property ratio (i.e. 0.44 on a scale ranging from 0 to 1);

Table 3

Schema and graph metrics with corresponding quality properties addressed and values recorded.

Metric	Description	Quality property	Value
Relationship Richness	The ratio between non-inheritance relations and the total number of relations defined in the ontology as proposed by [11]. Inheritance relations are `rdfs:subClassOf` axioms. Values range from 0 (i.e. the ontology contains inheritance relationships only) to 1 (i.e. the ontology contains non-inheritance relationships only).	Transparency	0.44
Inheritance Richness	The average number of subclasses per class computed as proposed by [11]. Inheritance Richness (IR) is expressed on ordinal scale. Its values should be interpreted relatively to the number of classes. If IR is much smaller than the number of classes, then the value is low. On the contrary, if IR tends to equalise the number of classes, the the value is high. A low value indicates a deep (or vertical) ontology, while a high value indicates a shallow (or horizontal) ontology.	Transparency	2.48
Axiom/class ratio	The ratio between axioms and classes computed as the average amount of axioms per class. Its values should be interpreted relatively to the number of classes and axioms. If the ratio is much smaller than the number of classes, then the value is low. On the contrary, if the ratio is much greater than the number of classes, the the value is high. Low values (i.e. ~ 0) indicate poorly axiomatised ontologies. On the contrary, higher values indicate better axiomatisations. Extremely high values might indicate over axiomatisation.	Transparency	39.55
Class/property ratio	The ratio between the number of classes and the number of properties. Typically good values are in the range $[0.3, 0.8]$ indicating a sufficient number of properties connecting things with other things (i.e. object properties) and values (datatype properties). Low values (i.e ~ 0) indicate an ontology with many properties connecting few concepts. On the contrary, high values indicate an ontology with many concepts connected by few properties. Nevertheless, the interpretation of the ratio always depends of the ontology size.	Cognitive ergonomics	0.44
Average Population (AP)	The average distribution of instances across all classes as computed by [11]. Namely, it computes the average number of instances per class in the knowledge graph. This metric provides an indicator about the extensional coverage of the ontology. Although there is not a clear positive or negative value, the metric should be interpreted relatively to the number of classes and individuals in the knowledge graph. For example, if the AP value multiplied by the number of classes returns a value that is much lower than the number of individuals, then the value can be interpreted as negative. In fact, in such a situation the knowledge graph contains many individuals that are not typed by any class defined in the ontology. Hence, the extensional coverage is low.	Compliance to expertise	58,914.53
NoR	The number of root classes as defined by [12]. A root class is a class that is not subclass of any other class in the ontology. NoR values are on ordinal scale and provide an indication of cohesion, i.e. the degree of relatedness between the different ontological entities. The interpretation of NoR values depends on the number of classes in the ontology. For example, 8 as NoR value might be low or high if the number of classes is 300 or 10, respectively.	Flexibility, Transparency	16
NoL	The number of leaf classes as defined by [12]. A leaf class is a class that has no sub-class in the ontology. NoL values are on ordinal scale and provide an indicator of cohesion, i.e. the degree of relatedness between the different ontological entities. Again, the interpretation of NoL values depends on the number of classes in the ontology. For example, 8 as NoL value might be low or high if the number of classes is 300 or 10, respectively.	Flexibility, Transparency	277
NoC	The number of external classes as defined by [14]. An external class is a class defined in a different ontology. Values for NoC are on ordinal scale. A low value of NoC suggests self-containment and semantic independence of an ontology. On the contrary, a high value suggests strong semantic dependency of an ontology with concepts defined in external ontologies. As for other metrics on ordinal scale the interpretation of good or negative NoC values is relative. For example, if the NoC is comparable to the number of internal classes then self-containment and semantic independence might not be guaranteed. In fact, a large portion of the ontology relies on concepts defined elsewhere. Accordingly, a change in an external ontology might affect the intended semantics deeply.	Flexibility, Transparency	38
Average breadth	The average breadth [13] computed on the graph whose nodes are ontology classes and edges are `rdfs:subClassOf` axioms. The metric suggests the degree of horizontal modelling (i.e. flatness) of the hierarchies of an ontology. Values are on ordinal scale. The value should be interpreted relatively to the number of classes. For example, average breadth values of 10 and 100 in an ontology consisting of 600 classes are low and high, respectively.	Cognitive ergonomics	5.75
Max breadth	The maximal cardinality recorded on ordinal scale over the graph constructed as for the average breadth [13]. The interpretation of max breadth is similar to that suggested for the average breadth.	Cognitive ergonomics	34
ADIT-LN	It records the average depth of the graph constructed as for the average breadth. The average is computed as the sum of the depth of all paths divided by the total number of paths [12]. ADIT-LN values are on ordinal scale and are indicators of cohesion. The interpretation of the values depends on the size of the ontology. Accordingly, low values occur when ADIT-LN is significantly lower then the number of classes. On the contrary, high values occur when the difference between ADIT-LN and the number of classes is less significant.	Transparency, Cognitive ergonomics	3.93
Max depth	The maximal depth obtained by traversing `rdfs:subClassOf` axioms in the graph constructed as for the average breadth. The interpretation of max depth is similar to that suggested for ADIT-LN	Cognitive ergonomics	5
Tangledness	The degree of multihierarchical nodes in the class hierarchy computed according to the formula provided by [13]. A multihierarchical node is a class having multiple super classes. Values for tangledness range from 0 (i.e. no tangledness) to 1 (i.e. each concept in the ontology has multiple super classes)	Cognitive ergonomics	0.56

Table 4
Results of the module metrics.

Ontology Module	Atomic size	Appropriateness	Encapsulation	Coupling
Denotative Description	6.37	1	0.99	0
Catalogue	5.64	1	0.99	0
Context Description	6.63	1	1	0
Core	4.85	1	0.96	0
Cultural Event	5.78	1	0.96	0
Location	5.50	1	0.99	0

- the low depth and breadth of the inheritance tree (i.e. 3.93 as ADIT-LN, 5 as max depth, 5.75 as average breadth, and 34 as max breadth);
- the moderate tangledness (i.e. 0.56 on a scalse ranging from 0 to 1);
- the use of patterns. With regards to this it is worth noticing that patterns identify dense areas within the knowledge graph. In fact, most of the top-ranked classes among the most instantiated (cf. Figure 15) identifies patterns, such as those described in Section 4.3. Significant examples are `roapit:TimeIndexedRole`, `a-cat:CatalogueRecordVersion`, and `a-dd:CulturalEntityTechnicalStatus` that count 2,758,760, 1,676,180, and 1,030,566 individuals, respectively;
- the high number of annotations (i.e. 8,374, which is much higher than the total number of classes and properties in the ontology) that facilitates user readability.

The average distribution of instances across all classes (i.e. average population) provides an indicator of the compliance to expertise from the structural perspective. In fact, we record a good extensional coverage with an average distribution of $\sim 59k$ individuals per class. We remind that this value is positively interpreted by following the interpretation for the AP metric defined in Table 3.

Module metrics suggest that all modules are modelled by following a similar design principles: identifying small and highly cohesive partitions as basic building blocks for ontology design. This result is fully compliant with the pattern-based approach adopted for modelling ArCo. As a matter of fact, the atomic size values we record are low and they differ only slightly from one module to another, i.e. ranging from 4.85 (Core module) to 6.63 (Context Description module). The appropriateness values recorded are optimal (=1 for all modules). In fact, the appropriateness value for a module ranges from 0 (i.e. no appropriateness) to 1 (i.e. complete appropriateness) [17]. We record excellent values for encapsulation (~ 1 for all modules). We remark that encapsulation values range from 0 (i.e. no encapsulation) to 1 (complete encapsulation). According to [16] a high encapsulation value is a good indication of the quality of a module. In fact, it suggests that such a module can be easily exchanged for another, or internally modified, without side-effects. The extremely low value for coupling (=0 for all modules) is excellent. Again, coupling values range from 0 (i.e. low coupling) to 1 (i.e. high coupling). Low coupling for an ontology module means that its entities do not have strong relations to entities in other modules. Accordingly, it is easy to modify and update such modules independently. Furthermore, the high encapsulation values along with the low coupling values suggest a high degree of independence of a module. This indicates that ArCo modules are self-contained and can be updated and reused separately. Thus, ArCo modules address the flexibility property identified by [13], which prospects an ontology/module that can be easily adapted to multiple views.

6.2. Logical and functional dimensions

Experimental setup and results. Logical and functional dimensions have been evaluated by means of unit testing [24] (cf. Section 4.5). For this purpose we define 18 test cases for inference verification, 29 test cases for error provocation, and 55 test cases for competency question verification. Each test case is publicly available on GitHub[49] and it is modelled by using the *testannotationschema*[52] ontology. For both inference verification and error provocation we define data samples to use with the HermiT reasoner for checking (i) the soundness of ArCo in inferring correct axioms (i.e. inference verification) and (ii) producing expected *in vitro* logical inconsistencies (i.e. error provocation). The aforementioned data samples are available on GitHub along with test cases.We rely on automatic reasoning as inference verification and error provocation provide an indication about the computational integrity and efficiency. [13] defines computational integrity and efficiency as the property that prospects an ontology that can be successfully processed by a reasoner. We use TESTaLOD for competency question verification by providing the corresponding test cases as input to the tool. The results obtained record that all test cases are successful. Competency question verification aims at validating the compliance to expertise from the functional perspective, i.e. the compliancy of an ontology with functional requirements representing its ontological commitments.

Additionally, we further analyse the functional dimension by setting up an experiment aimed at assessing ArCo ontologies with regards to their ability in capturing and conveying domain-specific terminology. This is of utmost important to further assess whether ArCo addresses its intended use, i.e. compliance to expertise. Inasmuch as only measuring the terminological coverage for ArCo might not be informative, we set up this experiment as a comparative analysis. For the comparison we select EDM and CIDOC-CRM as they are two well known and widely used ontologies in the same domain of ArCo. The terminological coverage is modelled as an ontology alignment problem between the vocabulary that represent the domain-specific terminology and the target ontology (ArCo, EDM, and CIDOC-CRM, respectively). The vocabulary is automatically extracted with Rapid Automatic Keyword Extraction [31] (RAKE) from a corpus composed of the ICCD cataloguing standards, their associated guidelines, and ArCo competency questions.

The resulting vocabulary counts of 55 terms and is publicly available as RDF[72]. The ontology alignment is computed with Silk [32] by using the *substring* metric with 0.5 as threshold. The alignment with the vocabulary is executed three times, i.e. once for each ontology involved in the comparison. The configuration files provided as input to Silk are available on FigShare[73].

Figure 16 reports the results of the terminological coverage for ArCo, EDM, and CIDOC-CRM.

[72]https://doi.org/10.6084/m9.figshare.7926599.v1

[73]The link specification files for ArCo, CIDOC-CRM, and EDM are published with the DOIs https://doi.org/10.6084/m9.figshare.7925555.v1, https://doi.org/10.6084/m9.figshare.7925573, and https://doi.org/10.6084/m9.figshare.7925867, respectively.

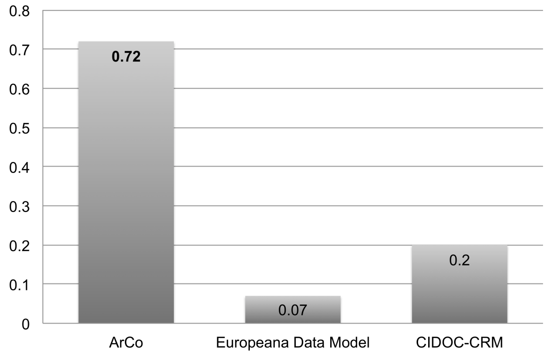


Fig. 16. The terminological coverage as recorded for ArCo, EDM, and CIDOC-CRM.

Discussion. The successful execution of inference verification, error provocation, and competency question verification is an indicator of (i) computational integrity and efficiency, and (ii) compliance to expertise. The former suggests that the ontology can be successfull processed by a reasoner. The latter suggests that ArCo is compliant with its collected requirements. Finally, the terminological coverage measured for ArCo (i.e. 0.72) shows a very good result and the comparison with the results obtained for the Europeana Data Model (i.e. 0.07) and for CIDOC-CRM (i.e. 0.2) further support the claim that such existing reference ontologies are not enough for addressing ArCo requirements.

7. Developing a KG using XD: lessons learned

This project led us to reflect on both strong and weak points of the methodology applied, thus suggesting possible improvements for the future. In particular, in this section we want to focus on two key aspects of eXtreme Design methodology: (re)using patterns and test-driven design. Finally, we discuss how involving the community let us collect a wider set of requirements.

7.1. Reusing existing ontologies and patterns

eXtreme Design is a methodology that encourages the reuse of Ontology Design Patterns (ODPs), as common modelling solutions to classes of problems recurring in ontology design. Patterns to be reused can be both selected from dedicated catalogues (such as the *ODP Portal*[46]) and extracted from state-of-the-art ontologies. In Section 4.4 we briefly explained the two main practices for ontology reuse: *direct* and *indirect* [29].

Even if ODP catalogues represent a relevant support for pattern-based ontology design, there is lack of well-documented and well-maintained high-quality ontology design patterns, as well as of tools for supporting ODP-driven ontology-engineering [33], which could guide the user in the selection of ODPs, e.g. by recommending possible ODPs to be reused for a certain modelling requirement. Additionally, using available ontologies as input to generate new ontologies is a difficult process, far from being automated [34], and can be hampered by scarsly documented ontologies, ontologies big in size and with a high number of classes, properties and axioms. Moreover, there is a need to carefully (thus time-consuming) consider the context, intended usage and semantic meaning of ontology entities. Issues in reusing existing ontologies seem to be confirmed by [35], which observes a lack of explicit alignments between ontological entities in Linked Open Data, while the high number of top level classes may suggest a high number of conceptual duplicates.

Ontology reuse would benefit from annotations about the ODPs implemented by ontologies: [30] proposes a simple representation language for ontology design patterns (OPLa ontology), which makes use of OWL annotation properties for documenting ODPs. OPLa certainly contributes to fill a gap but its expressiveness requires an improvement. ArCo has been annotated with OPLa, but we soon realised that we were missing many relevant attributes of, and relations between, patterns that could be annotated and therefore possibly later detected from other parties.

As described in Subection 4.3, during ArCo development we incrementally selected a CQ from the available list and then match it with one or more existing ODPs. We also inspected state-of-the-art ontologies, such as CIDOC-CRM[20], EDM[19], BIBFRAME[23], FRBR[24], etc., in this process. In all cases, this matching activity was incremental and manual, and a significant effort has been made to look for reusable fragments in big ontologies such as CIDOC-CRM. We believe there is a urgency in developing methods for automatically detecting ODPs used in ontologies as well as in building tools able to provide a modularised ODP-based visualisation of ontologies. These tools would help making the inspection of ontologies clearer and more understandable, hence easing ontology reuse, and contributing in supporting automatic matching procedures. Some work have considered detection of Ontology Design Patterns, e.g. [36] and [37]. Nevertheless, to the best of our knowledge, there is no automatic procedure able to recognise ODPs in knowledge graphs nor for annotating and reusing them yet.

7.2. Support for test-driven methodologies

Testing an ontology network, which is periodically released in unstable and incremental versions, can be a time-consuming and repetitive activity, and, if performed manually, error-prone. Tests need to be run in order to validate our ontology, by translating competency questions into SPARQL queries, verifying expected inferences and provoking expected errors. Each time there are changes over the ontologies (e.g. a new version which models new information), new tests are created, and all previous tests must be executed again and, if needed, updated, in order to identify new possible bugs.

While performing testing in the context of ArCo, we realised that tools automatising it would have been of great support for the testing team. Building TESTaLOD (described in Section 4.5.1) helped us executing tests over new versions of the ontology network, allowing for automatic regression tests. At the moment TESTaLOD only addresses CQs-based testing and their corresponding SPARQL queries. Tests for inference verification and error provocation are executed externally. Moreover, the creation and annotation of test cases is not automatised. We believe that developing tools supporting (semi-) automatic creation of unit tests is of paramount importance to push the overall quality of released knowledge graphs. TESTaLOD is just a scratch on the surface of a possible tool suite for automatising many activities of ODP-based and test-driven methodologies such as XD.

7.3. Extended customer team for Cultural Heritage LOD projects

In ontology engineering methodologies, domain experts are the main actor and input source of requirements and validation tests: they give a crucial contribution, especially in defining domain and task requirements that guide the ontology design and testing phases [38]. User stories (then translated into Competency Questions) were used as a *lingua franca* for making communication effective between ontology designers and ICCD domain experts, during the development of ArCo.

Whilst not denying the key role played by ICCD domain experts in eliciting requirements, by means of both cataloguing standards, catalogue records and discussions on specific topics and issues, we believe that the Cultural Heritage (CH) domain has a specificity in its users: the community interested in CH data for different purposes is wide and diverse, involving domain experts, researchers, art critics, students, simple citizens, institutions and companies owning and managing CH data or data on related domains (e.g. tourism), public administrations and private companies offering services related to the CH domain, etc.

Cultural Heritage is usually managed with a top-down approach, where professionals and data owners (Galleries, Libraries, Archives, Museums, etc.) are in charge of defining standards and means for describing, representing and making available data on cultural heritage. More rarely, end-users are involved in this process. Instead, institutions aiming at enhancing cultural heritage would benefit from a bottom-up approach, alongside a top-down one, for collecting requirements from the community that consumes their data.

Linked Open Data projects can help in getting domain experts closer to their potential wide and diverse audience, and in promoting interactions between them. In carrying out the ArCo project, considering the characteristics of the CH domain and CH users, we involved a wider community in the requirements and feedback collection phase. Launching an Early Adoption Program, and involving the community in the unstable and incremental phases of the project, allowed us to capture a wider range of perspectives and requirements. For example, Synapta[74], which reuses ArCo ontologies for representing musical instruments belonging to Sound Archives & Musical Instruments Collection[75] (SAMIC), needed information on musical heritage to be prioritised in the design of the network, while, based on ICCD requirements, this was previously given a lower priority (due to lack of data).

With ArCo EAP, we experimented the involvement of private and public organisations, and extended XD to this purpose by identifying a set of tools (web forms, mailing lists, GitHub issue tracker), and practices that could support collecting requirements from such a diverse community (webinars, meetups). We believe that collecting requirements from a very diverse community is relevant for the CH domain but can apply also in other contexts, hence methodologies and possible supporting tools shall consider this aspect, so far neglected to the best of our knowledge, among their key requirements.

8. Conclusion

This paper presents how ArCo, a knowledge graph of Italian Cultural Heritage (CH), has been developed, following the principles of the XD methodology. There are other valuable LOD resources containing and describing the Italian CH. Nevertheless, ArCo has a prominent role in this domain, not only because it injects in LOD high-quality data, extracted from the official institutional database of Italian Cultural Heritage (General Catalogue), but also because the expressiveness of its ontologies means that its LOD can be used by scholars and researchers, e.g. in humanities, to make new discoveries and find new patterns. The expected impact of ArCo on the general CH domain is motivated by a set of new requirements, addressed by its ontologies, which have been overlooked so far. These requirements emerged both from the richness of details provided by the General Catalogue records as well as from a growing community of consumers and producers of CH LOD.

ArCo can have an impact on the general Semantic Web community as well, since it is designed by following a robust methodology, based on the reuse of ontology design patterns, including extensive testing, detailed documentation and tutorial material, and formal evaluation: thus, it is a well-documented case study of the application of a methodology of ontology engineering (eXtreme Design), and can be used as a reference example by other researchers that are approaching knowledge graph engineering.

ArCo is still evolving and growing, and can be further improved and enriched. We plan to extend our ontologies, in order to model other aspects not addressed by the current version, e.g. some specific characteristics of naturalistic heritage, like slides and phials associated to an *herbarium*, the optical properties of a stone, etc. Being ArCo an evolving project, we keep encouraging our community to give us new requirements, in addition to continuous feedback, that we aim at addressing in the future. Moreover, in future requirement collection iterations, we want to extend our customer team to interested citizens, and further investi-

[74]https://synapta.it/

[75]http://museopaesaggiosonoro.org/sound-archives-musical-instruments-collection-samic/

gate how to best capture requirements from such a diverse audience.

ArCo LOD will be enriched by extracting structured data from many textual metadata contained in the catalogue records (e.g. generic narrative descriptions of the cultural properties, historical biographical data about authors, etc.), using NLP techniques. Additional effort is being put to complete the translation of the data to other languages, starting from English, generating automatic translations to be validated. Finally, ArCo has highlighted the need for tools for facilitating reuse and testing, in general but also specific to the CH domain.

References

[1] C. Bizer, T. Heath and T. Berners-Lee, Linked Data - The Story So Far, *International Journal of Semantic Web Information Systems* **5**(3) (2009), 1–22, DOI:10.4018/jswis.2009081901.

[2] C. Dijkshoorn, L. Jongma, L. Aroyo, J. van Ossenbruggen, G. Schreiber, W. ter Weele and J. Wielemaker, The Rijksmuseum collection as Linked Data, *Semantic Web* **9**(2) (2018), 221–230. doi:10.3233/SW-170257.

[3] V. de Boer, J. Wielemaker, J. van Gent, M. Oosterbroek, M. Hildebrand, A. Isaac, J. van Ossenbruggen and G. Schreiber, Amsterdam Museum Linked Open Data, *Semantic Web* **4**(3) (2013), 237–243. doi:10.3233/SW-2012-0074.

[4] E. Hyvönen, Semantic Portals for Cultural Heritage, in: *Handbook on Ontologies*, S. Staab and R. Studer, eds, International Handbooks on Information Systems, Springer, 2009, pp. 757–778. doi:10.1007/978-3-540-92673-3.

[5] A. Isaac and B. Haslhofer, Europeana Linked Open Data - data.europeana.eu, *Semantic Web* **4**(3) (2013), 291–297. doi:10.3233/SW-120092.

[6] M. Daquino, F. Mambelli, S. Peroni, F. Tomasi and F. Vitali, Enhancing Semantic Expressivity in the Cultural Heritage Domain: Exposing the Zeri Photo Archive as Linked Open Data, *JOCCH* **10**(4) (2017), 21:1–21:21. doi:10.1145/3051487.

[7] V.A. Carriero, A. Gangemi, M.L. Mancinelli, L. Marinucci, A.G. Nuzzolese, V. Presutti and C. Veninata, ArCo: The Italian Cultural Heritage Knowledge Graph, in: *The Semantic Web - ISWC 2019 - 18th International Semantic Web Conference, Auckland, New Zealand, October 26-30, 2019, Proceedings, Part II*, C. Ghidini, O. Hartig, M. Maleshkova, V. Svátek, I.F. Cruz, A. Hogan, J. Song, M. Lefrançois and F. Gandon, eds, Lecture Notes in Computer Science, Vol. 11779, Springer, 2019, pp. 36–52. doi:10.1007/978-3-030-30796-7_3.

[8] M. Doerr, The CIDOC Conceptual Reference Module: An Ontological Approach to Semantic Interoperability of Metadata, *AI Magazine* **24**(3) (2003), 75–92. doi:10.1609/aimag.v24i3.1720.

[9] E. Blomqvist, V. Presutti, E. Daga and A. Gangemi, Experimenting with eXtreme Design, in: *Proceedings of the 17th International Conference on Knowledge Engineering and Management by the Masses (EKAW)*, P. Cimiano and H.S. Pinto, eds, Lecture Notes in Computer Science, Vol. 6317, Springer, 2010, pp. 120–134. doi:10.1007/978-3-642-16438-5_9.

[10] E. Blomqvist, K. Hammar and V. Presutti, Engineering Ontologies with Patterns - The eXtreme Design Methodology., in: *Ontology Engineering with Ontology Design Patterns - Foundations and Applications*, P. Hitzler, A. Gangemi, K. Janowicz, A. Krisnadhi and V. Presutti, eds, Studies on the Semantic Web, Vol. 25, IOS Press, 2016. ISBN 978-1-61499-675-0. doi:10.3233/978-1-61499-676-7-23.

[11] S. Tartir, I.B. Arpinar and A.P. Sheth, Ontological evaluation and validation, in: *Theory and applications of ontology: Computer applications*, Springer, 2010, pp. 115–130. doi:10.1007/978-90-481-8847-5_5.

[12] H. Yao, A.M. Orme and L. Etzkorn, Cohesion Metrics for Ontology Design and Application, *Journal of Computer Science* **1**(1) (2005), 107–113. doi:10.3844/jcssp.2005.107.113.

[13] A. Gangemi, C. Catenacci, M. Ciaramita and J. Lehmann, Modelling Ontology Evaluation and Validation, in: *The Semantic Web: Research and Applications, 3rd European Semantic Web Conference, ESWC 2006, Budva, Montenegro, June 11-14, 2006, Proceedings*, Lecture Notes in Computer Science, Vol. 4011, Springer, 2006, pp. 140–154. doi:10.1007/11762256_13.

[14] A.M. Orme, H. Yao and L. Etzkorn, Coupling Metrics for Ontology-Based Systems., *IEEE Software* **23**(2) (2006), 102–108. doi:10.1109/MS.2006.46.

[15] A. Schlicht and H. Stuckenschmidt, Towards Structural Criteria for Ontology Modularization, in: *Proceedings of the 1st International Workshop on Modular Ontologies, WoMO'06, co-located with the International Semantic Web Conference, ISWC'06 November 5, 2006, Athens, Georgia, USA*, P. Haase, V.G. Honavar, O. Kutz, Y. Sure and A. Tamilin, eds, CEUR Workshop Proceedings, Vol. 232, CEUR-WS.org, 2006.

[16] M. d'Aquin, A. Schlicht, H. Stuckenschmidt and M. Sabou, Criteria and Evaluation for Ontology Modularization Techniques, in: *Modular Ontologies: Concepts, Theories and Techniques for Knowledge Modularization*, H. Stuckenschmidt, C. Parent and S. Spaccapietra, eds, Lecture Notes in Computer Science, Vol. 5445, Springer, 2009, pp. 67–89. doi:10.1007/978-3-642-01907-4_4.

[17] Z.C. Khan, Evaluation Metrics in Ontology Modules, in: *Proceedings of the 29th International Workshop on Description Logics, Cape Town, South Africa, April 22-25, 2016.*, M. Lenzerini and R. Peñaloza, eds, CEUR Workshop Proceedings, Vol. 1577, CEUR-WS.org, 2016.

[18] C. Dijkshoorn, L. Aroyo, J. van Ossenbruggen and G. Schreiber, Modeling cultural heritage data for online publication, *Applied Ontology* **13**(4) (2018), 255–271. doi:10.3233/AO-180201.

[19] P.A. Szekely, C.A. Knoblock, F. Yang, X. Zhu, E.E. Fink, R. Allen and G. Goodlander, Connecting the Smithsonian American Art Museum to the Linked Data Cloud, in: *The Semantic Web: Semantics and Big Data, 10th International Conference, ESWC 2013, Montpellier, France, May 26-30, 2013. Proceedings*, P. Cimiano, Ó. Corcho, V. Presutti, L. Hollink and S. Rudolph, eds, Lecture Notes in Computer Science, Vol. 7882, Springer, 2013, pp. 593–607. doi:10.1007/978-3-642-38288-8_40.

[20] E. Hyvönen, E. Mäkelä, M. Salminen, A. Valo, K. Viljanen, S. Saarela, M. Junnila and S. Kettula, MuseumFinland-Finnish museums on the semantic web, *Web Semantics: Science, Services and Agents on the World Wide Web* **3**(2–3) (2005), 224–241. doi:10.1016/j.websem.2005.05.008.

[21] M. Dragoni, E. Cabrio, S. Tonelli and S. Villata, Enriching a small artwork collection through semantic linking, in: *The Semantic Web. Latest Advances and New Domains - 13th International Conference, ESWC 2016, Heraklion, Crete, Greece, May 29 - June 2, 2016, Proceedings*, H. Sack, E. Blomqvist, M. d'Aquin, C. Ghidini, S.P. Ponzetto and C. Lange, eds, Lecture Notes in Computer Science, Vol. 9678, Springer, 2016, pp. 724–740. doi:10.1007/978-3-319-34129-3_44.

[22] P. Hitzler, A. Gangemi, K. Janowicz, A. Krisnadhi and V. Presutti (eds), *Ontology Engineering with Ontology Design Patterns - Foundations and Applications*, Studies on the Semantic Web, Vol. 25, IOS Press, 2016. ISBN 978-1-61499-675-0.

[23] M. Gruninger and M.S. Fox, The role of competency questions in enterprise engineering, in: *Proceedings of the IFIP WG5.7 Workshop on Benchmarking - Theory and Practice*, Trondheim, Norway, 1994, pp. 83–95.

[24] E. Blomqvist, A.S. Sepour and V. Presutti, Ontology Testing - Methodology and Tool, in: *Knowledge Engineering and Knowledge Management - 18th International Conference, EKAW 2012, Galway City, Ireland, October 8-12, 2012. Proceedings*, A. ten Teije, J. Völker, S. Handschuh, H. Stuckenschmidt, M. d'Aquin, A. Nikolov, N. Aussenac-Gilles and N. Hernandez, eds, Lecture Notes in Computer Science, Vol. 7603, Springer, 2012, pp. 216–226. doi:10.1007/978-3-642-33876-2_20.

[25] A. Gangemi and S. Peroni, The Information Realization Pattern, in: *Ontology Engineering with Ontology Design Patterns - Foundations and Applications*, P. Hitzler, A. Gangemi, K. Janowicz, A. Krisnadhi and V. Presutti, eds, Studies on the Semantic Web, Vol. 25, IOS Press, 2016, pp. 299–312. ISBN 978-1-61499-675-0. doi:10.3233/978-1-61499-676-7-299.

[26] A. Gangemi and V. Presutti, Multi-layered n-ary Patterns, in: *Ontology Engineering with Ontology Design Patterns - Foundations and Applications*, P. Hitzler, A. Gangemi, K. Janowicz, A. Krisnadhi and V. Presutti, eds, Studies on the Semantic Web, Vol. 25, IOS Press, 2016, pp. 105–131. ISBN 978-1-61499-675-0. doi:10.3233/978-1-61499-676-7-105.

[27] A. Gangemi, Norms and plans as unification criteria for social collectives, *Autonomous Agents and Multi-Agent Systems* **17**(1) (2008), 70–112. doi:10.1007/s10458-008-9038-9.

[28] V.A. Carriero, A. Gangemi, A.G. Nuzzolese and V. Presutti, An Ontology Design Pattern for representing Recurrent Events, in: *Proceedings of the 10th Workshop on Ontology Design and Patterns (WOP 2019) co-located with the 18th International Semantic Web Conference (ISWC 2019), Auckland, New Zealand*, K. Janowicz, A.A. Krisnadhi, M.P. Villalón, K. Hammar and C. Shimizu, eds, CEUR Workshop Proceedings, Vol. 2459, CEUR-WS.org, 2019.

[29] V. Presutti, G. Lodi, A.G. Nuzzolese, A. Gangemi, S. Peroni and L. Asprino, The Role of Ontology Design Patterns in Linked Data Projects, in: *Conceptual Modeling - 35th International Conference, ER 2016, Gifu, Japan, November 14-17, 2016, Proceedings*, I. Comyn-Wattiau, K. Tanaka, I. Song, S. Yamamoto and M. Saeki, eds, Lecture Notes in Computer Science, Vol. 9974, Springer, 2016, pp. 113–121. doi:10.1007/978-3-319-46397-1_9.

[30] P. Hitzler, A. Gangemi, K. Janowicz, A.A. Krisnadhi and V. Presutti, Towards a Simple but Useful Ontology Design Pattern Representation Language, in *CEUR Workshop Proceedings*, Vol. 516, CEUR-WS.org, 2017.

[31] S. Rose, D. Engel, N. Cramer and W. Cowley, Automatic Keyword Extraction from Individual Documents, in: *Text Mining. Applications and Theory*, M.W. Berry and J. Kogan, eds, John Wiley and Sons, Ltd, 2010, pp. 1–20.

[32] J. Volz, C. Bizer, M. Gaedke and G. Kobilarov, Silk-a link discovery framework for the web of data., in: *LDOW 2009*, Vol. 538, CEUR-ws, 2009.

[33] E. Blomqvist, P. Hitzler, K. Janowicz, A. Krisnadhi, T. Narock and M. Solanki, Considerations regarding Ontology Design Patterns., *Semantic Web* **7**(1) (2016), 1–7. doi:10.3233/SW-150202.

[34] M. Uschold, M. Healy, K. Williamson, P. Clark and S. Woods, Ontology reuse and application, in: *Proceedings of the 1st International Conference on Formal ontology in Information Systems*, Vol. 179, IOS Press, 1998, p. 192.

[35] L. Asprino, W. Beek, P. Ciancarini, F. van Harmelen and V. Presutti, Observing LOD using Equivalent Set Graphs: it is mostly flat and sparsely linked, in: *The Semantic Web - ISWC 2019 - 18th International Semantic Web Conference, Auckland, New Zealand, October 26-30, 2019, Proceedings, Part I*, C. Ghidini, O. Hartig, M. Maleshkova, V. Svátek, I.F. Cruz, A. Hogan, J. Song, M. Lefrançois and F. Gandon, eds, Lecture Notes in Computer Science, Vol. 11779, Springer, 2019, pp. 57–74. doi:10.1007/978-3-030-30793-6_4.

[36] M.T. Khan and E. Blomqvist, Ontology design pattern detection-initial method and usage scenarios, in: *SEMAPRO 2010, The Fourth International Conference on Advances in Semantic Processing*, M. Popescu and D.L. Stewart, eds, IARIA, 2010, pp. 19–24, DOI:10.1007/978-3-642-33876-2.

[37] A. Ławrynowicz, J. Potoniec, M. Robaczyk and T. Tudorache, Discovery of emerging design patterns in ontologies using tree mining, *Semantic Web* **9**(4) (2018), 517–544. doi:10.3233/SW-170280.

[38] G. Lodi, L. Asprino, A.G. Nuzzolese, V. Presutti, A. Gangemi, D.R. Recupero, C. Veninata and A. Orsini, *Semantic Web for Cultural Heritage Valorisation*, in: *Data Analytics in Digital Humanities*, S. Hai-Jew, ed., Springer, 2017, pp. 3–37. doi:10.1007/978-3-319-54499-1_1.

[39] C. Ghidini, O. Hartig, M. Maleshkova, V. Svátek, I.F. Cruz, A. Hogan, J. Song, M. Lefrançois and F. Gandon (eds), The Semantic Web - ISWC 2019 - 18th International Semantic Web Conference, Auckland, New Zealand, October 26-30, 2019, Proceedings, Part II, in *Lecture Notes in Computer Science*, Vol. 11779, Springer, 2019.