digitising contemporary art

DELIVERABLE

Project Acronym:	**DCA**
Grant Agreement number:	**270927**
Project Title:	**Digitising Contemporary Art**

D3.1 Metadata Implementation Guidelines for Digitised Contemporary Artworks

Revision: V1.1

Author(s): Sam Coppens (IBBT/MMLab), Erik Mannens (IBBT/MMLab)
External reviewers: Inge Van Nieuwerburgh (UGent), Patrick Hochstenbach (UGent)

Project co-funded by the European Commission within the ICT Policy Support Programme		
Dissemination Level		
P	Public	x
C	Confidential, only for members of the consortium and the Commission Services	

REVISION HISTORY AND STATEMENT OF ORIGINALITY

Revision History

Revision	Date	Author	Organisation	Description
V01	06/10/2011	BD, BL, RV	PACKED vzw	First internal review
V02	04/11/2011	SC, EM	IBBT	Amended draft
V03	09/11/2011	RV	PACKED vzw	Second internal review
V04	18/11/2011	SC, EM	IBBT	Amendments
V05	06/12/2011	IvN, PH	UGent	External review
V06	20/12/2011	SC, EM	IBBT	Amendments and additions
V1.0	21/12/2011	RV	PACKED vzw	Final version
V1.1	20/01/2012	RV	PACKED vzw	Final version (proofread by native English speaker Carmen Uranga)

Statement of originality:

This deliverable contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation or both.

CONTENT TABLE

Executive Summary

This deliverable D3.1 of the DCA project produces some guidelines for metadata of contemporary art. Based on the state-of-the-art in metadata in the field of contemporary art, these guidelines give an overview of the basic requirements for metadata creation (scheme, vocabulary, name convention, etc.) for cataloguing contemporary art. The guidelines also list all specifications (including an overview of metadata harvesting models) necessary for interoperability with Europeana and other aggregators.

The first chapter introduces metadata. What is metadata? Why is it needed? How is metadata serialised? What are the different sorts of metadata? These are all questions answered in the first chapter. Alongside such basic questions, the first chapter also introduces the different levels of digital resource (artwork vs. digital representation of the artwork) and gives a basic overview of the main metadata standards domain.

Chapter two stipulates guidelines for describing contemporary art for cataloguing purposes. It does this by first introducing a general model for describing artworks in general. These artwork resources typically need descriptions on the different levels of the digital resource. We therefore need both descriptions of the artwork itself and of the digital representation of the artwork. Next, this general model is applied to contemporary art to obtain the basic guidelines for describing contemporary art for cataloguing purposes.

Chapter three gives a detailed overview of some main metadata standards, together with their advantages and disadvantages. For the arts sector, CDWA and SPECTRUM are discussed. MARC is detailed for libraries and ISAD(G) and EAD for archives. In this chapter, EN15907 is also discussed in detail for describing audio-visual resources in general. Alongside these domain metadata standards for cataloguing purposes, some metadata standards are also discussed for metadata exchange purposes. Those that are introduced here are LIDO, Mets and OAI-ORE.

Chapter four is dedicated to the descriptions of some widely used vocabularies. It introduces the AAT vocabulary, a thesaurus for describing art and architecture. For place names, chapter four introduces the TGN vocabulary. For artist names, it introduces two vocabularies: ULAN and RKDArtists.

Chapter five produces some guidelines for metadata exchange in the field of contemporary art. For such, a horizontal as well as a vertical aggregator for contemporary art are reviewed. The horizontal aggregator is Europeana, and the Europeana data models ESE and EDM are considered. For the vertical aggregator, GAMA is mentioned, together with the GAMA model. GAMA is a harvester targeted specifically towards media art. The requirements for both aggregators are applied to LIDO in order to provide an application profile of it suitable for the metadata exchange of contemporary art. This assignment will be streamlined with the work that is being produced by the Linked Heritage project that is producing various profiles of LIDO, one of which is aimed at fine arts. The LIDO profile for fine arts will eventually become the main metadata exchange model for contemporary art, but meanwhile DCA's specific LIDO application profile is used.

Chapter six explains the strategy for how our DCA vocabulary might be used for the classification of contemporary art and its resources. For this vocabulary, the lexicons of the content partners are analysed in order to obtain a common subset of terms that are used by several institutions. This basic list is then extended with several terms from the GAMA vocabulary. This forms the basis for the DCA vocabulary. It is a work-in-progress and will be extended as needed. The extensions are discussed within a taskforce that will act as an editorial committee for the DCA vocabulary. In the end, it will also be mapped to AAT for interoperability and to anticipate the future use of this vocabulary in some aggregators.

The last chapter is dedicated to preservation metadata. DCA is a digitisation project. A large amount of digital resources representing artworks will be produced. Such a process already delivers a lot of the preservation metadata required to support the long-term preservation of digitised artworks. PREMIS is discussed in detail in this chapter as a metadata standard for preservation purposes.

1 Intro Metadata

1.1 What, why, when and how

Metadata is generally referred to as 'data about data'. A typical example of metadata are the catalogue cards that can be found in libraries, denoting the title of a book, the author, the date it was published, the publisher, where it can be found (catalogue number), etc. Metadata is usually textual information that describes the characteristics of a resource, e.g., its creation, content, or context.

Metadata can be provided in the form of free text, when it is intended only to be used by individuals, or in the form of structured textual information when it needs to be processed by machine agents. In the digital world, metadata is usually structured textual information. When the metadata is structured, it is made up of a number of pre-defined elements, representing the specific attributes of a resource. Each element can have one or more values. A metadata schema specifies the pre-defined elements and its possible values. This schema typically contains:

- A limited number of elements.
- The name of each element.
- The meaning of each element.
- The possible values of each element.

In short, metadata provides the means for us to describe our digital resources in a structured way in order to share them with other people and machines.

Metadata schema categories	Metadata vocabulary terms
Creator	Andy Warhol
Title	Marilyn Monroe
Subject	woman, portrait, contemporary art, ...

Figure 1: Metadata about a painting

These descriptions of resources serve different purposes. One important use of metadata is to locate a resource. For instance, a book reference is designed to give enough information to allow someone to retrieve that book. Another major use of metadata is resource discovery: finding resources of which one is unaware. These two metadata applications are referred to as search and retrieval. Information resources must be described in a way that allows end-users to tell whether the resources are useful or not. Metadata registration is a systematic method for describing resources and thereby improving access to them. If a resource is worth making available on the Web, then it is worth describing it with metadata to maximise the ability to locate it. Metadata provides the essential link between the information provider and the information consumer. While the primary goal of metadata is resource discovery and resource location, metadata schemas have been developed to support other functions:

- creation, multi-versioning, reuse, and re-contextualisation of information objects;
- organisation and description;
- validation;
- utilisation and preservation;
- disposition.

1.2 Different Sorts of Metadata

Metadata serves different purposes. It might be used to help find a resource (often termed 'resource discovery' metadata), or to tell us what that resource might be (descriptive metadata). It might tell us where the resource has come from, who owns it and how it can be used (provenance and rights metadata). It might describe how the digital resource was created (technical metadata), how it is managed (administrative metadata) and how it can be kept in the future (preservation metadata). Or it might, as mentioned above, help us relate this specific digital resource with other resources (structural metadata).

These are no distinct sets of metadata: there is obviously a considerable overlap. For example, descriptive metadata (e.g., subject of image) is also very important for searching and retrieving the image (resource discovery); while metadata relating to the creation of the resource (e.g., filename and format) is clearly also vital in managing and preserving it.

1.2.1 Binary metadata

Binary metadata describe the data on a bit level. Bit streams are the actual data in a file. Binary metadata, e.g., file system information and file header information, keep the enclosed information accessible by pointing out how the bits should be transformed to a representation of the data, e.g., in a certain format.

1.2.2 Technical metadata

Technical metadata describe the data from a technical perspective. It is metadata related to how a system functions or metadata behaves. Data formats and their derivatives evolve quickly. As file formats age, it is hard to find software that is still able to interpret old formats. The only way to keep this kind of information accessible is to support migration and/or emulation in which the technical metadata, e.g., coder-decoder (codec) information, is key.

1.2.3 Structural metadata

Structural metadata describe the relationship between a set of files that correspond to a possible representation of the intellectual content of certain data. An artwork can consist of various parts, for instance, an artist's book can consist of various drawings - and each drawing can be considered as an individual work. The structural metadata I define the relationship between the artist's book and its parts (drawings), detailing which drawings belong to the book and in which order they appear.

1.2.4 Descriptive metadata

Descriptive metadata is used to identify and describe collections and related information resources, e.g., artist, title, location, date, etc., to better find and locate the resource of interest.

When harvesting digital multimedia content from different cultural heritage institutions - whether museums, libraries, cultural institutions, or archives - an additional problem concerning descriptive metadata arises, i.e., many cultural heritage institutions describe, control, and save their descriptive metadata according to their own schemes. If an aggregator wants to file these extra descriptions as metadata, it is forced to choose one metadata standard to do so. This choice is not obvious, as most metadata schemes are domain specific. To guarantee lossless filing of all descriptive metadata the aggregator must opt for the lowest common multiple of all descriptive metadata schemes used by all content providers. This would lead to an enormous, unmanageable metadata scheme. We therefore suggest harvesting the descriptive metadata in its original metadata format, so as to be sure not to lose any information. The aggregator also foresees a generally accepted, descriptive and exchangeable metadata scheme (a greatest common divisor) to be effectively used to search the complete, non-homogeneous collection of harvested information resources. As such, the original metadata (saved as data itself) can be presented to the end-users, when the right (meta)data is found in the first place.

1.2.5 Administrative Metadata

Administrative metadata is metadata that describes everything needed to manage the data. It indicates, e.g., internal identifiers, when a resource was last modified and by whom, when it was created, etc. This metadata will describe all the information needed to support the organisational workflows that manage the information resources.

1.2.6 Preservation metadata

Preservation metadata describe essential extra data that support and document the digital preservation process. For such, it actually combines administrative metadata with technical metadata. No digital storage device is perfect and perpetually reliable, as bit preservation is still an unsolved paradigm. As stated earlier, information in a digital form is a conceptual object. This information can be altered and copied pretty easily without one notifying as such in its visible representation. Unlike analogue information, it is indeed much harder to preserve the authenticity of digital information. Adding tenability metadata to the preservation package of the archived essence can solve this. Such metadata have check sums, digital signatures, certificates, encryption, and cyclic redundancy control for indicating the data has not been altered without being documented. Furthermore, an archived dataset also needs its provenance documented. This type of preservation metadata (e.g., encoding software, version history, references to the original sources, etc.) describes the genesis of the intrinsic information, i.e., the original owners of the data, the processes determining the current form of that data, and all of its available, intermediate versions, as this information is vital in verifying all changes the data has experienced from genesis until date. Lastly, context-aware metadata (e.g., related data sets, help files, original language on first publication, etc.) must be retained, as these describe any possible relationship between the intrinsic data and other data that is not included within its own information package.

1.2.7 Rights metadata

Rights metadata describe the rights on digital objects, on descriptive metadata and on the original physical or digital resource/object. (e.g., rights metadata for describing copyright statements, (changing) licenses, and possible grants), as this information is also vital to guaranteeing long-term access to the data, and as such must also be archived . Rights metadata can be placed in a broader context and called ‘use metadata’. Use metadata manages user access, user tracking and multi-versioning information.

When developing a metadata schema for the long-term preservation of digital multimedia, metadata descriptions on all levels have to be taken into account, from bit level descriptions to those of the intellectual content.

1.3 Different Levels of Digital Resource

Metadata might focus on describing different levels of a digital resource. Although we will generally want to describe individual resources (e.g. a photo, a moving image or an audio file), sometimes we may prefer to describe aggregations of resources (e.g. a photo album, an online learning resource or a music album). Or perhaps we might wish to describe just a part of a larger whole (e.g. an illustration found within a published book, a particular scene from a moving image file or a single track of music from an audio file). Metadata standards have approached this challenge in different ways. Some have created separate metadata records to describe individual 'things' (e.g. collection, single item, part of an item) and then made links within the metadata record to related files and metadata records, e.g. the Dublin Core (DC) schema (1). Some have created complex metadata schemas that are capable of describing different levels within a single metadata record, e.g. the SEPIADES schema (2). Others use different kinds of metadata to describe the various levels of a complex resource and then tie them together using special metadata schemas that are intended to structure and coordinate other metadata, e.g. the METS schema (Metadata Encoding and Transmission Standard, (3)).

As well as being focused on different levels, metadata might describe different 'layers' of content within the digital resource. Take again for example the Marilyn Monroe painting by Andy Warhol. In this case there might be (a) an original artwork (the painting), (b) a photographic reproduction of that artwork (a slide), and (c) a digital representation of that work (a digital file). The table below shows how the metadata might differ according to the different 'content layer' being described.

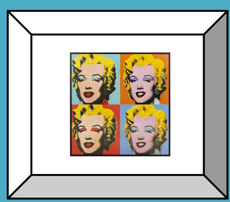
			010010100101 010010100101 001010101010 100001001010 101010101001 010100101010 011001010101
	Original image	**Slide image**	**Digital image**
Creator	Andy Warhol	Jan Smit [Photographer]	Davy Dekkers [Scanning Technician]
Format	Painting	Photographic transparency	JPEG image
Location	Andy Warhol Museum	University slide collection	A:\images\0023.jpg

Figure 2: Metadata about a painting, a slide and an image

1.4 Different Metadata Encodings

Encoding schemes define how one's metadata is transformed into a textual description in order to be processed by machines. Important schemes include:

- HTML (Hyper-Text Markup Language, (4))
- XML (eXtensible Markup Language, (5))
- RDF (Resource Description Framework, (6))

In this section, we elaborate on the different encoding schemes that are currently in use.

1.4.1 From Html to XML to RDF: A Little History

Nowadays, the hypertext Web is a fact. It is actually a web of documents. These documents are described using the Hypertext Mark Up Language, or HTML. HTML is a language specially designed to describe web pages and the links between them. Such a webpage usually consists of a body of text interspersed with multimedia objects, e.g., images, interactive forms, or movies. HTML provides a means to describe the structure of text-based information in a document. It is able to denote text as links, headings, tables, etc. This text is supplemented with embedded images, interactive forms and other objects. These HTML pages can be consulted using HTML browsers, e.g., Mozilla Firefox, which can present a webpage in a readable form for people viewing.

```
<html>
<body>
<p>
An image:
<img src="http://example.com/MarilynMonroe.gif"width="144" height="50">
</p>
<p>
Painter: Andy Warhol
</p>
</body>
</html>
```

A lot of data presented on web pages comes from (relational) databases, spreadsheets, address books, etc. HTML is not built to describe such data. It can only describe a webpage, which is a visual representation of that data. For describing data, the eXtensible Markup Language, or XML, was designed. XML is a set of rules for representing and structuring data in a textual format. In the same way that HTML uses tags and attributes to describe a webpage, XML uses tags to describe a piece of data. XML-parsers use these tags to extract the right piece of data from an XML document.

```
<document href="http://example.com/MarilynMonroe.gif">
  <painter>Andy Warhol</painter>
</document>
```

The next evolution was the eXtensible HyperText Markup Language, or XHTML. XHTML restricted the rules of HTML to those of XML. It is actually a reformation of HTML to XML. This made it possible for XML-parsers to parse XHTML documents, or to map XML documents automatically to XHTML representations. The benefits of XML-based Web documents (i.e. XHTML) involve searching, indexing and parsing as well as future-proofing the Web. XML was a driving force behind the disclosure of a lot of (meta)data stored in databases, spreadsheets, technical drawings, etc.

And yet, XML still has some interoperability issues. The same piece of information in the previous example can be described in XML as:

```
<painter>
   <uri>http://example.com/MarilynMonroe.gif</uri>
   <painter>Andy Warhol</painter>
</painter>
```

Or as:

```
<document href="http://example.com/ MarilynMonroe.gif" painter=" Andy Warhol " />
```

These XML documents all describe the same piece of information, which is obvious for a person reading them. However, for a machine parsing these XML documents, the documents all produce different XML trees. This makes querying over the XML tree very difficult and syntax dependent. Furthermore, the tags used in the XML document do not mean anything for a machine. For a person the tags already give a hint as to what their semantic meaning may be. This makes exchanging information using XML a non-trivial task.

A solution for this is the Resource Description Language, or RDF. RDF describes information using triples. These triples consist of a subject (“http://example.com/MarilynMonroe.gif“), a predicate (“painter”), and an object (“Andy Warhol”). Using these triples, any piece of information can be described by an RDF graph, which consists of a set of triples. These RDF graphs can also be described in a textual, interchangeable format, e.g., RDF/XML (7), Turtle (8), etc. When a machine parses such textual descriptions, they all end up with the same RDF tree. This is the task of RDF reasoners, which build up the RDF tree, and makes querying its syntax independent. Furthermore, all the nodes of the tree get a semantic meaning in RDF. For this RDF introduced namespaces. Namespaces are Uniform Resource Identifiers (URIs). By appending a namespace to the “XML tags”, those “tags” become unique, which makes it possible to define the semantics of that “tag”. A tag with a namespace forms the predicate in RDF. This allows for an easy exchange of information, a reuse of information and reasoning over that information.

An extension of RDF is the RDF Schema (9), or RDFS. RDFS is very similar to the XML schema: it describes the structure of the RDF document, and defines the semantics of its elements. It allows structuring data with classes and properties on those classes. Another extension of RDFS is the Web Ontology Language, or OWL (10). OWL extends RDFS by introducing more descriptive logics. This allows, for instance, for it to be stated that “all paintings by Andy Warhol are contemporary art”, even if that information is not included in the description of the painting.

With such techniques, the Semantic Web is emerging in today's world. The machine-readable descriptions enable content managers to add meaning to the content, i.e. to describe the structure of the knowledge we have about that content. In this way, a machine can process knowledge itself instead of text. It can use processes similar to human deductive reasoning and inference, thereby obtaining more meaningful results and helping computers perform automated information gathering and research.

1.4.2 eXtensible Markup Language (XML)

XML is an extensible mark-up language used to describe any piece of information on a textual basis. An XML document allows for information to be represented and processed by machines. XML is very similar to HTML, but is intended to describe a piece of information, not to describe a web page in a visual manner.

XML documents are composed of mark-up and content. There are six kinds of mark-up that can occur in an XML document: elements, entity references, comments, processing instructions, marked sections, and document type declarations.

An XML document begins with a processing instruction. While it is not required, its presence explicitly identifies the document as an XML document and indicates the version of XML to which it was authored.

An XML document is structured by elements. Elements are the most common form of mark-up. Delimited by angle brackets, most elements identify the nature of the content they surround. Some elements may be empty, as seen above, in which case they have no content. If an element is not empty, it begins with a start tag, <element>, and ends with an end tag, </element>. XML is extensible, so it allows for one to define one's own elements, with one's own tags to denote the element. The elements can contain other elements, which support the structuring of one's XML document.

Besides elements, attributes are also used for mark-up. Attributes are name value pairs that occur inside tags after the element name. For example, <div class="preface"> is the div element with the attribute class having the value preface. In XML, all attribute values must be quoted.

XML also introduces namespaces. Namespaces are used to convert one's element tags into URIs. URIs are just identifiers for an element. When a URI is resolvable, which means that you can look it up in a browser, it becomes a URL. This is because it not only identifies one's element, but also locates where to find information on that element. Namespaces are used to disambiguate used elements and to group elements together that relate to a common idea.

Some basic rules for XML documents are:

- XML is case sensitive.
- All start tags must have end tags.
- Elements must be properly nested.
- XML declaration is the first statement.
- Every document must contain a root element.
- Attribute values must have quotation marks.

```xml
<?xml version="1.0"?>

<metadata
 xmlns="http://example.org/myapp/"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xsi:schemaLocation="http://example.org/myapp/ http://example.org/myapp/schema.xsd"
 xmlns:dc="http://purl.org/dc/elements/1.1/">

 <dc:title>
  For the Love of God
 </dc:title>
 <dc:description xml:lang="en">
  For the Love of God is a sculpture by artist Damien Hirst
  produced in 2007. It consists of a platinum cast of a human skull
  encrusted with 8,601 flawless diamonds,
  including a pear-shaped pink diamond located in the forehead.
 </dc:description>
 <dc:creator>
  Hirst, Damien
 </dc:creator>
 <dc:date>
  2007
 </dc:date>
<dc:identifier>
  http://example.org/ForTheLoveOfGod/xml
 </dc:identifier>

</metadata>
```

XML declaration

Root element defined by the base namespace xmlns="http://example.org/myapp/"

start tag

end tag

Tags nested in the root tag <metadata>

Attribute denoting the language

An XML document can be associated to an XML schema. An XML schema will define the structure (elements and ordering of the elements) and the data types (the data types of the content of an element). The document together with the XML schema allows XML documents to be converted into a hierarchical structure, which enhances the machine processability of the XML documents. As an example, the XML schema of Dublin Core can be found on the following link: http://dublincore.org/schemas/xmls/qdc/2008/02/11/dc.xsd

XML documents can be well-formed, meaning they follow the syntactic rules of the XML specification, but can also be valid. A valid XML document conforms to its XML schema, meaning that it is first of all well-formed, that its structure conforms to the defined structure in the XML schema and that its data types correspond to the defined data types in the XML schema.

1.4.3 Resource Description Framework (RDF)

The Resource Description Framework (RDF) is a general-purpose language for representing information on the Web. It is a foundational standard for the Semantic Web. It allows for the representation of information in an unambiguous and machine-processable way.

As the name “Resource Description Framework” suggests, it is a framework that allows for the description of resources. A resource can be anything: a web page, a person, an idea, etc.

In RDF, a resource is identified by a URI. Formally, a URI is a Unicode string that:

- Does not contain any control characters (#x00 - #x1F, #x7F-#x9F)
- Would produce a valid URI character sequence (per RFC2396, sections 2.1) representing an absolute URI with optional fragment identifier when, subjected to the encoding described below.

In practice, however, a URI is often used as a resource identifier.

The underlying structure of RDF data is a collection of triples. Every triple consists of a subject (S), predicate or property (P) and object (O) respectively. A set of such triples is called an RDF graph.

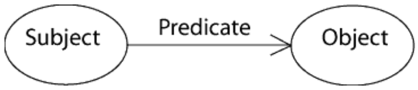


Figure 3: Conceptual representation of a triple

Note that the object of a triple can also be a literal instead of a URI. However, literals cannot appear as the subject or predicate of a triple. By convention, a literal is represented as a box instead of an ellipse.

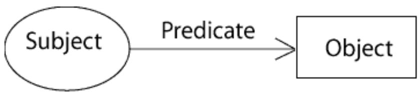
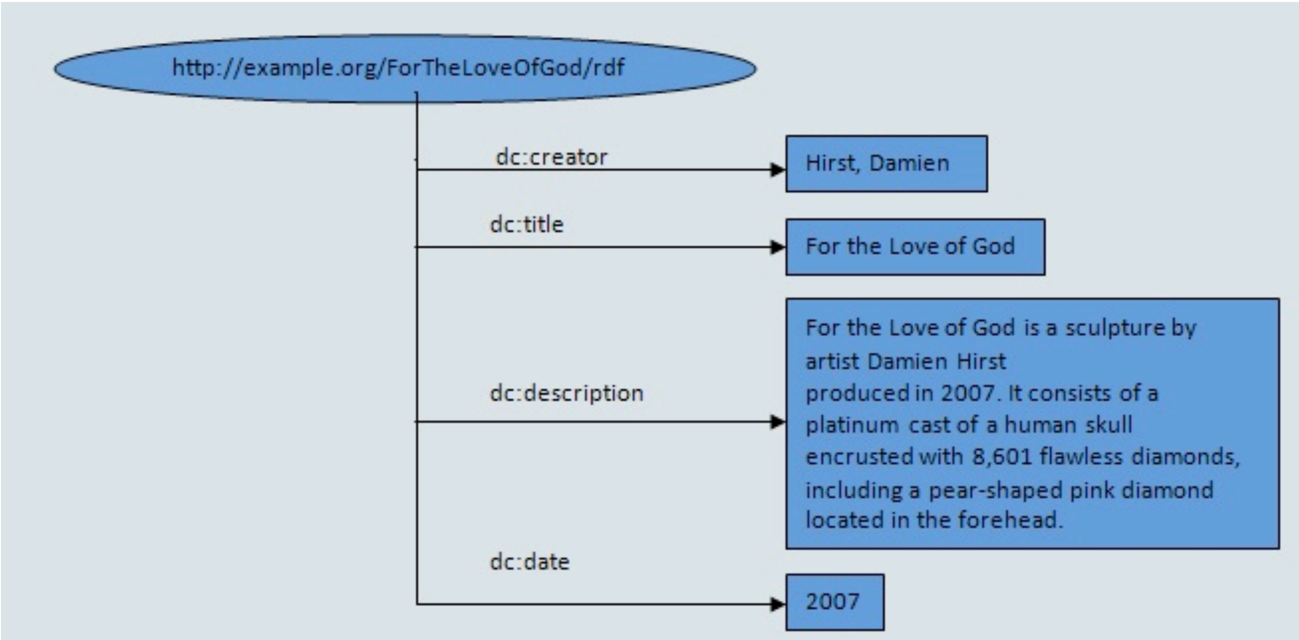


Figure 4: Triple with a literal as its object

All literals have a lexical form being a Unicode string, which should be in Normal Form C. A literal can also be provided with a language tag. Alternatively, a data type URI can be provided to a literal, forming a typed literal.

If we take the example of Damien Hirst's 'For the Love of God' record and put it in RDF, the record becomes a graph, represented below.



RDF can be serialised in XML for exchange of data. Below, one can find the example expressed in RDF/XML.

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
      xmlns:dc ="http://purl.org/dc/elements/1.1/">
 <rdf:Description rdf:about="http://example.org/ForTheLoveOfGod/rdf">
  <dc:title>For the Love of God</dc:title>
  <dc:description>For the Love of God is a sculpture by artist Damien Hirst
           produced in 2007. It consists of a platinum cast of a human skull
           encrusted with 8,601 flawless diamonds,
           including a pear-shaped pink diamond located in the forehead.</dc:description>
  <dc:date>2007</dc:date>
  <dc:language>en</dc:language>
  <dc:creator>Hirst, Damien</dc:creator>
  <dc:title xml:lang="fr">L'Initiative de métadonnées du Dublin Core</dc:title>
 </rdf:Description>
</rdf:RDF>
```

1.4.4 Terse RDF Triple Language (Turtle)

RDF has a recommended XML serialisation. However, for readability purposes, other representations such as the Terse RDF Triple Language (Turtle) exist. In Turtle, triples are represented as a sequence of terms (subject, predicate, object), separated by whitespace and terminating with '.' after each triple. URIs are written enclosed in '<' and '>'.

In order to further improve readability and provide shorter notations, URIs may also be abbreviated by using Turtle's '@prefix' directive. Literals are written either using double-quotes when they do not contain line breaks like "simple literal" or """long literal""" when they contain line breaks.

Language tags for literals are indicated by appending the literal with '@' and the language tag. Typed literals are represented by appending the literal string with '^^' followed by a data type URI. Often multiple triples are present that have the same URI subject in common. Also, triples with the same subject and predicate in common occur frequently. Turtle provides a more compact notation for these cases. The ',' symbol is used to repeat the subject and predicate of triples that only differ in the RDF object term. The ';' symbol is used to repeat the subject of triples that vary only in predicate and RDF object terms.

In this document, URIs will mostly be abbreviated using the '@prefix' directive. An example of a record description using the Turtle serialisation is shown below:

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>.
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>.
@prefix owl: <http://www.w3.org/2002/07/owl#>.
@prefix dc: < http://purl.org/dc/elements/1.1/>.

< http://example.org/ForTheLoveOfGod/rdf >
                    dc:title            “For the Love of God”;
                    dc:date             “2007”;
                    dc:description      “For the Love of God is a sculpture by artist Damien Hirst
                                        produced in 2007. It consists of a platinum cast of a human skull
```

```
                        encrusted with 8,601 flawless diamonds,
                        including a pear-shaped pink diamond located in the forehead.";
        dc:language     "en";
        dc:creator              "Hirst, Damien".
```

1.4.5 RDFS and OWL

RDF Schema (RDFS)

RDF provides a way to state facts or make assertions about resources, using named properties and values. Users also need to be able to define a vocabulary to use in such statements. For example, it should be possible to define and describe classes and properties to be used in them. RDF provides no such means. However, the RDF Schema (RDFS) vocabulary description language, another W3C Recommendation alongside RDF and Turtle, provides a vocabulary that can be used for this purpose.

The RDF Schema is defined in the form of an RDF vocabulary. The resources in this vocabulary have URI references with the prefix 'http://www.w3.org/2000/01/rdf-schema#' (conventionally associated with the prefix 'rdfs'). For a complete overview of RDF Schema, please refer to (9).

Web Ontology Language (OWL)

The OWL Web Ontology Language, also a W3C Recommendation, is designed for use by applications that need to process the information content instead of just presenting it to viewers. OWL enables a higher level of machine interpretability of content than is supported by RDF and RDF Schema, by providing an additional vocabulary along with formal semantics.

OWL is used to formally define and describe the meaning of the terms in a vocabulary and the relationships between the terms. This description is also called ontology.

Note that as of October 27th 2009, the OWL 2 Web Ontology Language (also known as OWL 2) has become a W3C Recommendation. OWL 2 is compatible with the OWL standard (now also often referred to as "OWL 1").

The RDF Vocabulary Description language (RDFS) and the Web Ontology language (OWL) provide a data modelling language for data on the Web. Semantics defined in the RDF Schema and OWL specifications allow RDFS- and OWL-reasoners to reason and consequently entail additional information from RDF data containing RDFS and OWL constructs.

1.5 Typology of Metadata Standards

Metadata serves several purposes, by which the far most important is easing the retrieval of relevant information. It may also help to organise electronic resources, to ensure resource interoperability, to provide resource digital identification, and to support resource archiving and preservation. There are currently a lot of metadata standards and choosing one has become far from easy. Metadata standards differ in purpose, the levels of digital resources they describe, and even in used encoding.

In general, one can divide metadata standards into several categories. There are **data content standards** (cataloguing rules and codes). These are guidelines for the format and syntax of the data values that are used to populate metadata elements. Examples of such standards are Anglo-American Cataloguing Rules (AARC, (11)), Resource Description and Access (RDA, (12)), and International Standard Bibliographic Description (ISBD, (13)). These standards do not provide a metadata model, or an encoding of the metadata model. Another category of metadata standards is the **data structure standards**, e.g., MARC21, CIDOC-CRM (14). These are "categories" or "containers" of data that make up a record or other information object. **Data format/technical interchange standards** are metadata standards expressed in machine-readable form. This type of standard is often a manifestation of a particular data structure standard, encoded or marked up for machine processing. A last category of metadata standards is the **data value standards** (controlled vocabularies, thesauri, controlled lists). These are the terms, names, and other values that are used to populate data structure standards or metadata element sets.

If a data structure standard does not have an encoded form, it is often referred to as a **conceptual standard**. These standards are meant to use as guidelines for building one's own internal metadata model for the management of information resources. Examples of such standards are FRBR (15), ISAD(G) (16), etc.

Some data structure standards or their encoded form (data format/technical interchange standard) are meant for cataloguing/registration purposes (**cataloguing metadata standards**), others are meant to exchange or publish information (**exchange metadata standards**). The former standards may require a very extensive set of metadata, while the latter might do with a more limited set. For cataloguing purposes information loss is unacceptable, whereas it might be acceptable for exchanging or publication purposes. On the other hand, metadata for exchange must be encoded in some way before it can be transferred, while cataloguing metadata standards do not always have an encoding. Examples of exchange metadata standards are MARCXML (17), and LIDO (18).

1.6 Different Domains, Different Data Structure Standards

Below, one can find a summary of the most widely used metadata standards in the cultural sector, libraries, and the archive sector, i.e., based on the domain they are used in or developed for. This classification is not arbitrary. The different sectors show quite an overlap, as some metadata standards can be applied in different sectors, yet there are some important differences. The museum and library world are more focused on describing individual information objects, while the archive world is more focused on describing the hierarchy / relations between groups of information objects. This distinction has major consequences, especially if the different data need to be mapped to a common scheme. The next chapter will discuss the metadata standards mentioned in detail.

The most common and probably the most simple metadata standard to use for publication or exchange purposes is **Dublin Core** (1). It's almost the lingua franca of metadata standards in terms of interoperability. It is not meant to be used as a cataloguing model. It is far too concise for that. It is a **cross-domain** metadata standard. The strength of this standard is its simplicity and generality. The standard consists of (only) 15 fields. These fields can describe any source, but its description is often too limited. Therefore, Dublin Core is widely used as an additional metadata standard alongside another metadata standards that describes sources much more precisely. Since most systems have knowledge of Dublin Core, a mapping of the standard metadata to it just provides the necessary interoperability. Because the 15 Dublin Core fields are optional and repeatable, almost any metadata standard can be mapped to it sometimes with some information loss, (not all fields can be mapped to the 15 Dublin Core fields).

Most important for the **museum** field, is **Categories for the Description of Works of Art** (CDWA) (19). It describes the data from art databases, using a conceptual framework for the description and retrieval of information about art works, architectural works and other cultural material. CDWA contains 512 categories and subcategories. A small subset of these categories composes the core, i.e., the minimum information needed for a work to be described and identified. This core is implemented as an XML schema, called CDWA Lite. This model is also consistent with OAI-PMH standard, which simplifies the exchange of data between various libraries. The OAI-PMH standard is a protocol for exchanging metadata between a content provider and content aggregator.

CIDOC-CRM (14) is another standard that is commonly used within the cultural sector. The CIDOC Conceptual Reference Model (CRM) provides definitions and a formal structure to describe the concepts and relationships used in the documentation of cultural heritage assets. Although it is a conceptual model, several encodings of it exist already, e.g., XML or OWL. CIDOC CRM focuses mainly on the description of contextual information, i.e., adding historical, geographical and theoretical background information of the exhibits drastically increases the value and significance of the original art pieces. This standard is often used as a metadata hub, cf. Dublin Core, in order to increase the interoperability of systems. Where Dublin Core does this with quite a bit of data loss, CIDOC-CRM (extensive as it is) is able to deliver interoperability with minimal data loss.

Another standard used by the museum field is **SPECTRUM** (20). It is a standard/conceptual model developed through collaboration with the museum sector. It describes a set of procedures for managing the collection as well as a set of information units or data the museum has to collect during such procedures. Using SPECTRUM ensures that a museum delivers excellent standards and meets its public responsibility.

Within the museum sector **Light Information Describing Objects** (LIDO) (18) is a widely used exchange metadata scheme. It is specifically targeted towards metadata exchange in relation to museum objects, so its focus is on descriptive metadata. It cannot be used as an internal cataloguing scheme, because it lacks, e.g., administrative metadata. It is based on standards like CDWA Lite, SPECTRUM, and CIDOC-CRM. It can therefore handle many different descriptive metadata models used by the museum sector.

Within the **library field**, **MAchine Readable Cataloguing** (MARC)/MARC21 (17) and **Functional Requirements for Bibliographic Record**s (FRBR) (15) are the most commonly used metadata standards. MARC is a standard for the representation and communication of bibliographic and related information. The standard's main function is therefore to simplify and speed up the retrieval of books in a library. The MARC format provides a high degree of granularity, which makes it quite complex. On the other hand, this granularity ensures that the source can be described very accurately indeed. The format is concise, as field names such as "place of publication", are replaced by a short code - which hampers the readability of the standard for people of course, but favours machine processability.

FRBR, on the other hand, is a mere conceptual model used in the library world, but with more emphasis on the end user. This model was developed to facilitate certain user activities, such as retrieval of records. Bibliographic entities defined in this model are divided into groups, which can in turn be further subdivided. The main characteristic of FRBR is that it comprises groups of entities:

- Group 1 entities are work, expression, manifestation, and item. They represent the products of intellectual or artistic endeavour.
- Group 2 entities are person and corporate body, responsible for the custodianship of Group 1's intellectual or artistic endeavour.
- Group 3 entities are subjects of Group 1 or Group 2's intellectual endeavour, and include concepts, objects, events and places.

This standard is also very granular and thus allows for a very accurate description of the entities. However, it should be noted that the use of highly granular metadata standards comes with extra implementation costs.

Within the **archival** sector the **ISAD (G) standard** (16) is often used. This standard assists in preparing descriptions of collections and their objects. The standard consists of several 'rules', but does not provide its own encoding scheme and should therefore be considered as a 'guide' for describing collections and their objects. These rules are, e.g., guidelines for the use of multi-level descriptions, references, titles, dates, etc.

A popular metadata standard for archives is EAD (21), the **Encoded Archival Description**. This is a metadata standard developed by the library of the University of California, Berkeley. The scheme consists of 146 elements to describe a collection as a whole, but also to describe different collection levels (parts of collections of even archival objects). This model can be seen as an encoding scheme of ISAD (G). The popularity of the standard is also due to its interoperability with other standards, like MARC or Dublin Core.

For **audio-visual archives**, **EN15907** and **EN15744** are recommended. They are European standards developed to describe cinematographic works, but they can be used in the broader context of audio-visual material. They aim not only to describe audio-visual works, but also to exchange descriptions of the works.

All aforementioned metadata standards are discussed in detail in Chapter 3.

1.7 Data Value Standards

In addition, metadata standards can be grouped according to search capabilities. Besides regular metadata standards (such as CDWA / CDWALite) there also exist semantic metadata standards which support 'intelligent' search methods that take into account the exact meaning of the search terms or that make use of already defined thesauri. These thesauri, i.e., dictionaries with 'agreed' terms, can thus overcome typical problems where computers are not able to see the resemblance of 'Andy Warhol', 'Warhol Andy' and/or 'A. Warhol'. Thesauri can therefore also help in searches by adding synonyms to the request.

There exist several vocabularies that are already widely used. Using such a vocabulary has several benefits. As explained before, it supports **information retrieval** because it can handle **synonyms**. It can cover even **spelling mistakes** during search. It can also support **multilingual search** or a **faceted search** as well as **categorisation**.

There exist thesauri for artist names, geographic places, sorts of animals, medical diseases, etc. For the cultural heritage sector, the most common thesauri are:

- AAT: A vocabulary for art and architecture, (22).
- TGN: A vocabulary for geographical places, (23).
- ULAN: A thesaurus for artist names, (24).
- LCHS: A thesaurus for subject headings maintained by the Library of Congress, (25).
- RKDArtists: A Dutch vocabulary for artist names, (26).

These vocabularies are discussed further in detail in Chapter 4.

2 Guidelines for Contemporary Art Metadata

In this section, we will provide some best practices for cataloguing contemporary art. These guidelines are applicable for the internal use of metadata at institution level. A separate chapter (Chapter 5) will provide guidelines for exchanging metadata, i.e., exporting metadata. We will focus first of all on the specific metadata requirements for art museums (and other art collecting institutions) and then apply them to contemporary art.

2.1 Metadata for art museums

Art collecting institutions require specific features for their metadata. The digital resource of an art museum has many more levels of description than, for instance, digital resources belonging to libraries.

Based on FRBR (15), a digital resource will consist of the following basic elements:

- Work: This element applies to an abstract concept, before it is made physical.
- Expression: An expression of a work is a representation of that work, but it is not yet a physical representation.
- Manifestation: A manifestation is a physical/digital representation of an expression. Thus, a manifestation can be a set of physical things, containing multiple expressions of work.
- Item: An item is a single example of a manifestation.

Although FRBR was originally designed for library objects, its conceptual model is also applicable for museum objects or archival objects. Museum objects differ from library objects, because they are unique, whereas library objects (usually) are not. Therefore, a museum object is characterised by one work, expression, manifestation and item, which are all unique. For libraries, a work and expression can (usually) have multiple manifestations and items. Archives, on the other hand, contain grouped objects, or collections, not individual objects. Archivists describe their archive hierarchically, first describing a collection then, on a lower level, its parts, and on down to the description of each individual object. For an archive, the collection of objects and the objects themselves are considered unique, because it archives a particular ‘item’ of a ‘work’. Because the nature of the objects are not the same for libraries, museums, and archives, library metadata standards, museum metadata standards and archival metadata standards also differ from each other.

Let us take the example of a library object, i.e., a book containing different poems. Every poem is a work of its own. The grouping of the poems is also a work of its own. This work will have whole-part relationships to the work of the individual poems. The publication of the book of poems is a manifestation, which includes expressions of the work of the entire group of poems, and expressions of each poem individually. An item, in this case, will refer to a specific, physical book.

Let us apply this model to an artwork and its related documents. Every physical thing conforms to a FRBR manifestation or item. When a manifestation is unique, it is also an item. This is the case for an artwork. The same holds true for all the documents related to the artwork. It can be, e.g., a slide, an analogue photo, a digital image, a certificate, a book, etc. All these related documents are also manifestations or items, because FRBR also considers digital resources as physical things. This leads to the schema, Fig. 5, which maps the artwork and its related documents to the FRBR model.

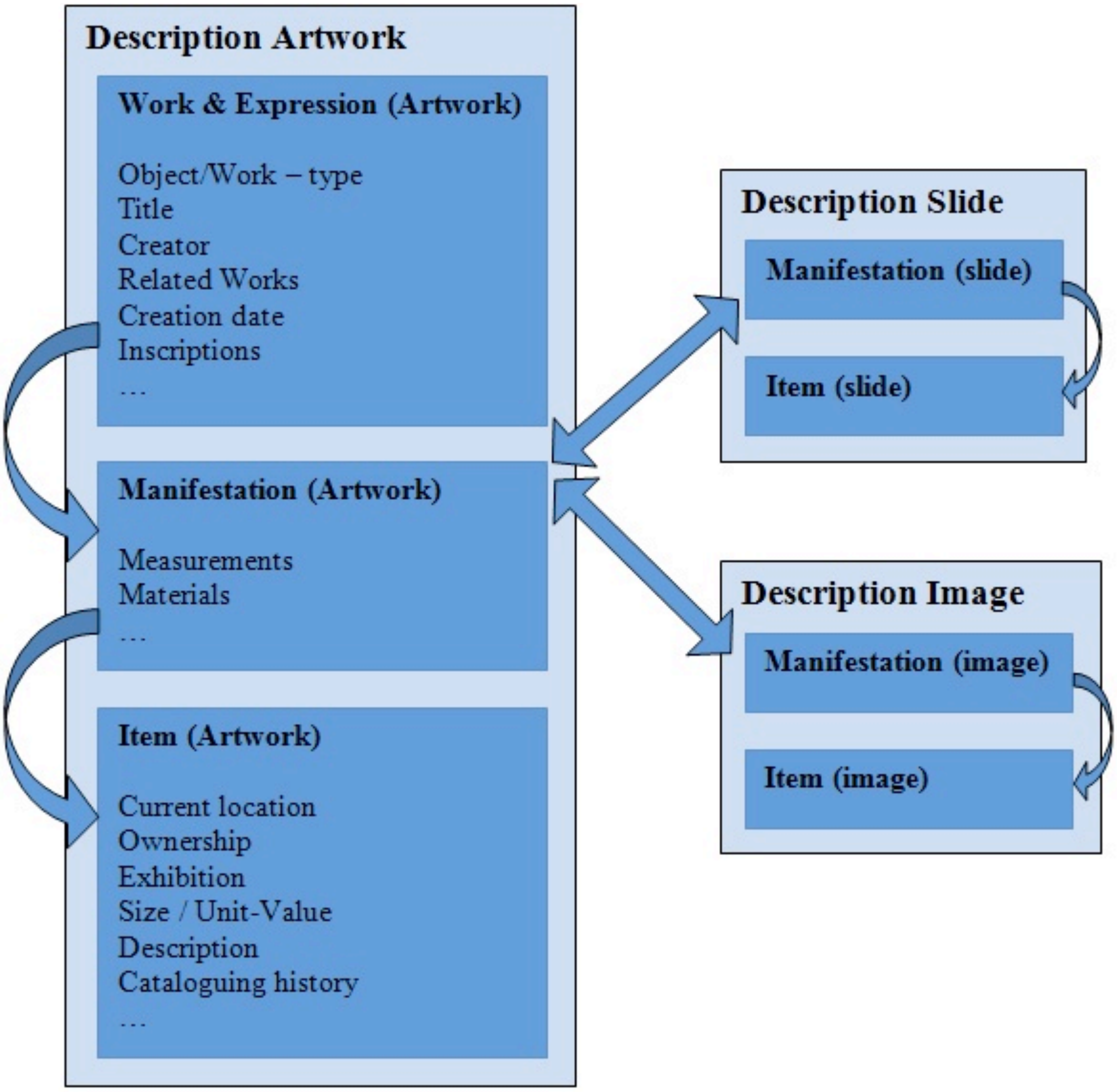


Figure 5: FRBR for artworks

This conceptual framework appears in several standards, like LIDO and EN15907. LIDO contains several elements. The root element is the object. An object contains descriptive metadata and administrative metadata. The descriptive metadata will describe the object itself. The administrative metadata contains links to records and resources. The record element contains administrative metadata about the metadata of the object and references to other metadata records. The resource element contains metadata about physical representations of the object. These physical representations can also be digital representations. The EN15907 has the same characteristics. They can thus be interpreted as practical implementations of the FRBR conceptual model.

2.2 Metadata for Contemporary Art

The basic elements that make up an object of contemporary art are in general:

- The artwork itself
- Digital or analogue representations of that artwork (e.g. digital reproductions of the artwork)
- Related documentation (e.g. certificates, a plan of the installation)
- Context information (information of involved agents, places, dates, events, etc.)

The figure below gives a schematic overview of a contemporary art object.

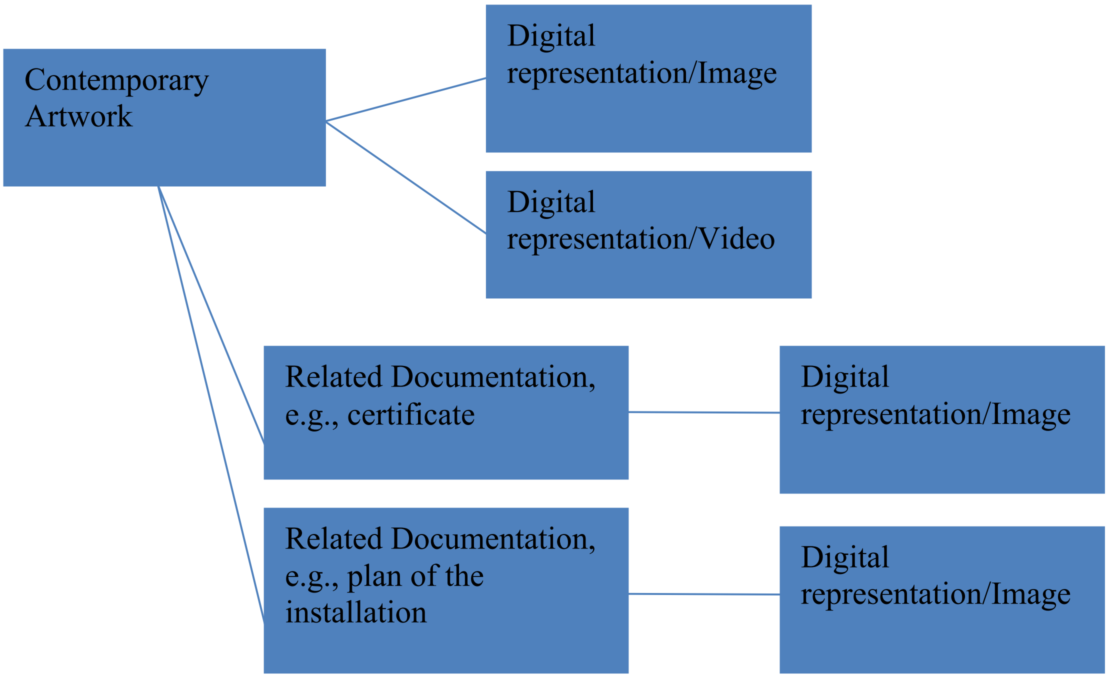


Figure 6: Schematic overview of a contemporary art object

There are three objects now: the contemporary artwork, the certificate and the plan of the installation. Each of these objects needs descriptive metadata. These objects can be related to each other in the record element of the administrative element as explained above. Every object has its own resources (digital representations), which are related to their object (artwork, certificate or plan of the installation). These resources need to be described too. Thus, we end up with three analogue objects, which need to be described. Each analogue object can have a digital representation (surrogate), which also needs a description.

Every object is different in nature and thus has its own specific requirements. One has to make a distinction between:

- The artwork and its related documentation.
- The object (artwork or related documentation) and its representation.
- Analogue objects and digital objects.

The artwork, the related documentation and their representations are different objects, i.e., they will need different descriptions. Artworks are unique, while their documentation may not be (e.g., a book). An artwork and a representation of the artwork (e.g., a digital photo of the artwork) will also need different descriptions. Consider the type element of a descriptive metadata record. The type in the descriptive metadata of the artwork object will refer to the type of the artwork itself, e.g., a painting. The type of the representation will denote if the resource is a video, image, sound recording, etc. The creator of the artwork is the artist; the creator of the representation is the photographer, the video maker or sound recordist. The distinction between analogue and digital objects also has consequences on the metadata of the object. While metadata records of analogue objects will have information on the measurements or the materials used, the metadata records on digital objects will contain technical metadata like, e.g., the file format of the digital object, or its size in terms of bits and bytes.

One can also opt to develop one's own metadata schema for implementing these metadata descriptions and their relations, but this might have some disadvantages:

- Existing tools for certain metadata standards cannot be reused.
- Metadata export will require mappings to the standards, because no one understands the metadata model.
- Metadata standards incorporate a lot of domain knowledge, because domain experts developed them. Using one's own metadata model can result in the loss of some best practices.

A better solution would be to compile one's own set of metadata standards that can be used for describing objects and resources. First, one has to describe the artwork itself. For this, museum standards like CDWA, CIDOC-CRM and SPECTRUM are the most suitable. If the artwork is digital, one must also include technical information in the artwork description. Good candidates for this are EN15907 and even CDWA, CIDOC-CRM or SPECTRUM extended with PREMIS.

For the related documentation, one can use MARC, EAD, ISAD(G) or also CDWA, CIDOC-CRM and SPECTRUM. Choosing a specific metadata standard must be based on the characteristics of the relating documentation. A detailed discussion of the metadata standards mentioned follows in Chapter 3, to guide the decision making of the appropriate standard. If the relating document is for instance a book, MARC is an obvious choice. If it is a hand-drawn version of the artwork, it can be considered an artwork of its own and CDWA is a good candidate. If the related documentation comes from a collection of documents, EAD would be the recommended standard because it is also able to describe the collection, apart from the document itself. If the documentation is a digital resource, the description must also contain technical metadata. A good candidate for digital resources is EN15907, or one of the aforementioned standards, extended with PREMIS information.

These metadata standards can then be extended to include some institution-specific metadata elements, e.g., to denote when an artwork was last checked. These extensions, however, must use their own namespace. It is not recommended to reuse the namespace of the standard that one is extending. As such, one can tailor the standards to one's institution's needs, incorporate all the domain knowledge of the metadata standards used, and be able to reuse existing tools for them, e.g., mapping tools.

Special attention needs to be given to the technical metadata of digital representations. DCA is a digitisation project. Many artworks will therefore be digitised. This digitisation process already provides all the technical metadata you need for describing digital representations in a more technical way. Such technical metadata is very valuable and should not be thrown away, but rather incorporated in the metadata during the whole digitisation process. The technical information provided can then be used to complete the EN15907 descriptions or the PREMIS object descriptions. Of course, not all technical information provided should be taken for granted, and it should be validated before being included in the descriptions.

Particularly for contemporary art, a lot of effort needs to go into relating all objects to each other. An artwork can consist of several other individual artworks; an artwork may have lots of documentation, cfr. certificates, and an artwork must have context information related to it. The latter means that the description of the artwork or its digital representation should be related to descriptions of persons, events, places, etc. For instance, a record of a person with their name, date and place of birth will provide a lot more context information than just including a string in the description with the name of the person. The record of a person can disambiguate between persons with the same name, whereas a string is not able to do so. The same holds for places. Paris is the name of several places in the world, but when you have a record on Paris, with its geographical coordinates, this allows disambiguation between all places called Paris in the world. The more relations to persons, places, or events you have, the more context information you provide to the end user. All objects (artwork, documentation, context, and artwork representations) must be properly related to each other. Relating the different objects to each other, one can extend the standards used with one's own vocabulary for relating all the objects, or use the OAI-ORE model for specifying the relationships, or reuse certain metadata fields from metadata standards (e.g., CIDOC-CRM) for relating the different objects.

Once one has compiled one's own set of metadata standards for describing the artwork and its related documents, the metadata fields need to be completed. For this, there exist several standards. First of all, there are the data content standards (cataloguing rules and codes). These are guidelines for the format and syntax of the data values used to populate metadata elements. Examples of such a standard are AARC (12), RDA (13), and ISBD (14). They provide information on how a title, name or date should be provided in metadata fields. Next, there are the data value standards (controlled vocabularies, thesauri, controlled lists). These can be used for some fields of the metadata descriptions. The appropriate data value standards have to be selected for populating certain metadata fields. Candidates are ULAN or RKDArtists for denoting artists, TGN for denoting geographical places, and AAT for categorising artworks. These controlled vocabularies are discussed in detail in Chapter 4.

2.3 Operational steps for Cataloguing Contemporary Art

1) Describe the artwork using CDWA, CIDOC-CRM or SPECTRUM. If the artwork is digital, extend the aforementioned descriptions with PREMIS in order to include the necessary technical metadata or use EN15907, which already includes the technical metadata.
2) Describe the related documents. For this, standards from any cultural sector may fit (libraries, arts sector, and archives), depending on the nature of the related document. Candidates are CDWA, CIDOC-CRM, SPECTRUM, MARC, EAD, ISAD(G). If the related document is digital, technical metadata must be included at this point. This can be done by extending the chosen metadata description with PREMIS or by using the EN15907 standard.
3) Representations of both the artwork and its documentation are considered related documents, so the same recommendations are valid .One also has to make a distinction between analogue (e.g., an analogue photograph, a painting) and digital representations (e.g., a digital photographic reproduction, a digital video recording).
4) For relating the different descriptions, one should compile one's own set of relating metadata fields and extend the existing descriptions with them. Another solution is using OAI-ORE for specifying one's own relating metadata fields or reusing existing metadata fields from other metadata standards (e.g., CIDOC-CRM).
5) Once these metadata standards are selected for describing the artworks and their related documents and representations, the metadata descriptions need to be filled in. For this

several data content standards exist, e.g., AARC, RDA, and ISBD.

6) Choose some controlled vocabularies for the enrichment of the metadata descriptions. Possible candidates are AAT, TGN, ULAN, and RKDArtists.

3 Data Structure Standards

3.1 CDWA and SPECTRUM for museums and the art sector

Categories for the Description of Works of Art (CDWA, (19)) is used to describe artworks and data about works of art, architecture, other material culture, groups and collections of works, and related images. The CDWA standard contains 512 categories and subcategories. The core consists of a small subset of these categories. These core categories include the minimum information needed to describe and identify a work of art. Alongside this core, CDWA also incorporates the possibility to include discussions, basic rules for cataloguing, and examples.

The CDWA standard is a product of the ART Information Task Force (AITF), which acts as an agent between art historians, art information professionals, and information providers. As such, they provide guidelines to describe art works, architecture, groups of art objects, and visual and textual surrogates of these art assets. This group was founded in the early 1990's and consisted of representatives of several communities that provide and use information about art, i.e., museum curators, professionals in the field of visual resources, art librarians, information managers and technical specialists. CDWA's categories provide a framework in which existing information can be mapped and upon which new systems can be developed. Furthermore CDWA identifies descriptive vocabularies and applications that make information in different systems more compatible and more accessible. Using the CDWA framework contributes to the integrity and longevity of the data, and will make the migration of data to new systems in the future easier. Above all, it helps the end user in the search for reliable information, regardless of the system in which data is stored.

CDWA proposes a relational data structure, where records about objects/works are linked together via hierarchical relationships. CDWA also recommends keeping track of related visual works in separate files, and doing the same for related textual material, persons, company information, locations, and the like. Such authoritative information on persons, places, concepts, and other topics can be important information to be found, but is better kept in separate files, and then integrated with the core information on an artwork. The advantage is that information should be described only once, and can thus be reused in the records of other artworks as well. CDWA Lite is an XML-schema that covers the core elements to describe a work of art or cultural material. It is based on CDWA and CCO. CCO stands for Cataloguing Cultural Objects, a descriptive standard also developed by the Getty. CDWA Lite records contribute to the unification of catalogues and libraries that use the OAI-PMH protocol for exchanging data. CCO, on the other hand, delivers a number of guidelines for selecting, ordering and formatting data used to further complete catalogue records. It therefore uses information related to a subset of both the CDWA categories and the VRA Core Categories. VRA Core is another metadata standard intended for visual cultural heritage information. The standard also provides a set of categories for images about the artwork.

Advantages:

- Developed specifically for works of art so that fields do not need to be duplicated and their meanings are obscured.

Disadvantages:

- Much more complicated than Dublin Core.
- Does not work well for describing surrogates of artworks.

CDWA is better suited for art collections than for (digital) visual resource collections. For contemporary art this means that CDWA can be used to describe the artworks themselves, but not for the digital representations of the artwork or contextual documents of the artwork.

SPECTRUM (20) is a British open standard that has been developed with the help, experience, and insights of 300 museum professionals. This standard describes procedures for documenting, handling and identifying objects. Further attention is also paid to, e.g., rights management, lending and risk management. SPECTRUM is constantly evolving and therefore offers the opportunity for growth and expansion. This standard is seen as the ‘industry standard’ for documenting works of art.

SPECTRUM consists mainly of two parts. Firstly, it describes a set of procedures, shown through the use of flowcharts, the main management processes in the museum. They can be used to review and improve practice in any size of museum. SPECTRUM Procedures are simple workflows that show how museums systems and processes can be used to manage key functions such as acquiring new material, assessing risk, managing copyright or making and receiving loans. SPECTRUM Procedures are not a rigid structure that museums have to follow. Instead, they provide a framework that allows the museum to respond to the needs of its collections and audiences.

Secondly, SPECTRUM describes a set of information units that museums need to collect during the management process to support public access, accountability and efficiency.

Advantages of SPECTRUM:

- SPECTRUM is generic. It applies to all museums regardless of their size or collection types.
- SPECTRUM is very practical. The procedures and information units are targeted towards the museum sector.
- More specifically for DCA, SPECTRUM is a good standard, but it is only a conceptual model. Many standards are already SPECTRUM compliant and incorporate the prescribed information units of SPECTRUM.

Disadvantages of SPECTRUM:

- SPECTRUM is generic. It does not make a distinction between the different sorts of collection a museum may hold.
- SPECTRUM is quite extensive. No museum has ever implemented all 21 described procedures.

3.2 MARC21 for libraries

MARC is an acronym for Machine Readable Cataloguing, (17). It is a standard for the representation and communication of bibliographic and related information - in a machine-readable form. The standard is maintained by the Library of Congress and has its origins in the 1960s as a digital form of library record sheets. The main purpose of the standard is:

- To ease the cataloguing of books by using copycat (two or more libraries exchanging metadata records)
- To ease the exchange of authority files
- To ease the creation of union catalogues.

As such, MARC data elements form the basis of most library catalogues. Furthermore, there is no alternative that delivers a similar degree of granularity. MARC supports eight types of 'material', including 'sound recordings' - that includes all types of audio (except music) -, 'computer file' - as a digitised version of, e.g., an oral historical source -, and the 'manuscript (textual) language material' type - as the transcription of, e.g., an oral history source.

MARC contains data elements for the following types of material:

- Books (printed, electronic, manuscript and microform textual materials)
- Continuing resources (textual materials issued in parts with a recurring pattern of publication)
- Computer files (computer software, numeric data, computer-oriented multimedia, online systems or services)
- Maps (printed, electronic, manuscript, and microform cartographic materials)
- Music (printed, electronic, manuscripts, and microform music, as well as musical or other sound recordings)
- Visual materials (projected media, non-projected media, two-dimensional graphics, three-dimensional graphics, naturally occurring objects)
- Mixed materials (mixture of material forms)

A typical MARC (bibliographic) record consists of multiple fields. There are fields for author, title information, etc. These fields can be further divided into subfields. The textual names of the fields (e.g., author and subject) are replaced with tags that consist of a three-digit code. These codes describe what kind of data is within this field.

Subfields are separated by a character (e.g., $, _, $$), which is complemented by a subfield code indicating what kind of information will follow. Two indicators further define some fields. A simple example of a typical MARC entry is:

```
245 10 $aInterview Andy Warhol$h[sound recording].
260 ## $aPittsburgh$bTheFactory$c1999.
300 ## $a1 minidisc$bdigital, ATRAC, stereo.
500 ## $aInterview with Andy Warhol.
500 ## $atranscription available.
511 0# $aInterview taken by X
```

The first line contains a field with the code '245', which indicates a 'Title Statement'. The indicators have the value 1 and 0 and the field contains subfields $a -the actual title-, and $h -the medium-. To make it easier to understand and manipulate the records, a MARC XML schema was designed. The example shows the high degree of granularity the MARC format provides, but also its associated complexity. The format is still concise. Field names, such as, 'place of publication' are in fact replaced by a short code.

MARC has no notion of semantic search, which means there's a search for the given keywords in all the different fields, but it does not take into account the real meaning or the concept of those keywords. Besides the bibliographic records that discuss the characteristics of resources, there are other types of records that, e.g., further describe their classification, give extra information about names, subjects, etc.

Advantages of MARC/MARC21:

- High granularity

- Widespread
- Can be written in an XML-syntax

Disadvantages of MARC/MARC21:

- Complex and hard to map
- No hierarchical structure
- No semantics
- Not suitable for "layman"

3.3 ISAD(G) and EAD for archives

A standard commonly used within the archive sector is ISAD(G) or General International Standard Archival Description, (16). This standard assists in preparing the descriptions of collections and their objects. The standard is merely a set of rules that must be followed. For multi-level descriptions (as possible within EAD) the advice is to fill in a description that is from general to more specific and clearly locate each description within the hierarchy. In a similar way, there are rules on how to complete references, titles, dating, etc. SEPIADES (SEPIA Data Element Set) is derived from ISAD(G). This specific standard typically aims at describing and managing photographic collections. It contains 21 'core' elements, and another 400 complementary elements. SEPIADES uses a multi-level approach similar to ISAD(G). These standards do not provide their own encoding schemes and are thus merely 'guidelines' for describing collections and their assets.

Advantages of ISAD(G):

- It is generic and is able to describe all sorts of archival data. It is a good guidance in designing metadata models

Disadvantages of ISAD(G):

- ISAD(G) looks at cataloguing practices, but does not provide a formal model or scheme that implements them

There exist several formal models that all claim to be ISAD(G)-compliant, but they all implement the standard from their own viewpoint and this creates interoperability issues. Cataloguing software, like Adlib or ICA-Atom claim to be ISAD(G) compliant, but this does not imply they are interoperable. ISAD(G) is a good guidance for designing metadata models, but it cannot be used to harvest data. One formal model closely related to ISAD(G) and a metadata standard is EAD.

EAD is an acronym for Encoded Archival Description. It is a metadata standard developed by the library of the University of California at Berkeley. It wanted to introduce/include more information than MARC records provide. Their extra requirements included:

- Possibility to provide extensive and interrelated descriptive information
- Ability to maintain the hierarchical relationships between the different levels of description
- Ability to provide descriptive information independent of these inherited hierarchical levels
- Ability to navigate within such a hierarchy of structured information
- Support for element-specific indexing and navigation

EAD (21) can be encoded in XML. The elements that may be used to describe an asset collection and how these elements can be ordered (e.g., what elements are needed, which elements are allowed in others, etc.) are specified in the EAD Document Type Definition (DTD). The tag set consists of 146 elements and is used both to describe a collection as a whole, and for the encoding of a detailed multi-level inventory of that collection. Many EAD elements can easily be mapped to other standards, such as MARC and Dublin Core. This increases both the flexibility and interoperability of data sets. An EAD consists of several parts:

The EAD header contains the title and detailed information about the collection and the document. The elements in the header are often mapped to Dublin Core elements. The archival description consists of the Data Item Description (DID) and is supplemented with any additional descriptions, and - what makes up the vast majority of information - the complete inventory of the collection.

The DID describes the collection as a whole, including the administrator (person or organisation), the language, a brief description, etc. This DID may be followed by several additional elements:

- A biographical description of the person or organisation
- A detailed description of the collection
- Description of objects related to the collection
- Objects belonging to the collection, but are separated from the collection (e.g., needing special handling, special storage requirements, ...)
- A list of topics or keywords for the collection
- Restrictions on the material in the collection.

The inventory of the collection is divided into progressively smaller pieces with more 'refined' information, thus allowing for the required information depth when searching and/or cataloguing. The Research Libraries Group also offers a 'coordinating centre' as a service. Members may submit their information to this group, which will then index the data and generate a search interface to the index. This allows researchers to search with a single query through hundreds of collections. An example of an EAD file is:

```
<filedesc>
<titlestmt>
<titleproper> Interview with Andy Warhol
<date>1962</date>
</titleproper>
<author>The Factory</author>
</titlestmt>
<notestmt>
<note> <p> <subject>Marilyn Monroe</subject> </p> </note>
</notestmt>
</filedesc>
```

Advantages of EAD:

- Can be translated into Dublin Core and/or MARC21 and many other standards
- Hierarchical structures are possible

Disadvantages of EAD:

- Not user friendly
- Different interpretations of the standard are possible

3.4 EN15907 and EN15744 for audio-visual collections

EN15907 (27) is a European standard developed to describe cinematographic works. It provides a set of metadata for the description of cinematographic works, but also a terminology to exchange descriptive metadata.

The standard consists of several entities related to each other for describing the works themselves, but also their variants, manifestations and items. The primary elements in the standard are:

Cinematographic Work:

> This is in fact the root element in the description. It will relate the work to all its variants and manifestations. It comprises both the intellectual and artistic content and the realisation in a cinematographic medium. A cinematographic realisation of a pre-existing non-film is also considered a cinematographic work. This means that concerts, original theatre performances, sport events, etc. can also be described as cinematographic works.

Variant:

> This entity is used to describe statements about the content-related characteristics that may vary without changing the cinematographic work. Such variants can be produced by minor additions, deletions or substitutions to the descriptions of the cinematographic work, as long as they do not lead to a 'new cinematographic work'.

Manifestation:

> A manifestation is the physical embodiment of the cinematographic work or one of its variants. Many cinematographic works are distributed online, without a physical carrier. This entity also refers to computer files.

Item:

> An item is a single exemplar of a manifestation. This also includes fragments of the manifestation. In the case of purely digital media, an item is also defined as the availability of the computer file from an owner, irrespective of the number of backup copies that might exist.

Content:

> Statements about the content of a cinematographic work can be made using one of the following elements:
>
> - Subject Terms:
>
>   This includes items from a concept scheme that denotes the subject, genre or form of the cinematographic work.
>
> - Content Description:
>
>   This element is used to store a textual description of the contents of a work. A content description can refer to a single piece, but it can also refer to several, as long as a complex piece exists that includes them.

Agent:

> Agents are involved in the creation, realisation, curation or exploitation of the cinematographic work, variant, manifestation or item. An agent can be a person, corporate body, family or a group of persons. An agent can be described by the inclusion of an authority record, a link to an authority record or by his/her name.

Event:

> An event is an entity that plays a role in the lifecycle of the cinematographic work, variant, manifestation or item. The following event types have been defined:
>
> - Publication Event: A public screening, broadcast, release online or on a physical medium.
> - Decision Event: A decision about the suitability of the work for a certain audience (includes censorship and rating decisions).
> - IPR Registration: The registration of the IPR.

- Award: An award relating to the cinematographic work.
- Production Event: An event in the creation of the work.
- Preservation event: The creation of a new manifestation or variant intended to backup the contents of a work.

The schema below gives an overview of the relations between the several entities defined by the EN15907 standard.

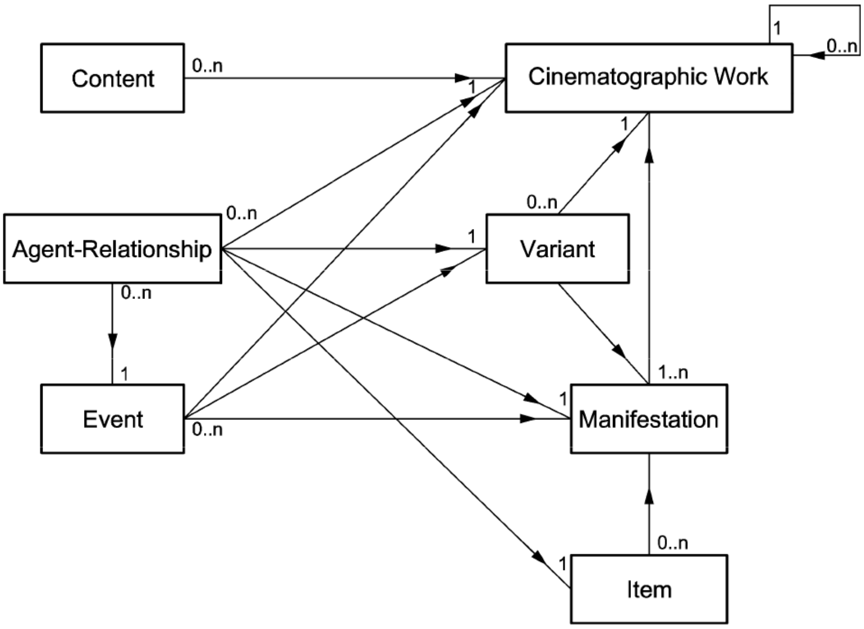


Figure 7: Conceptual model of EN15907

Advantages of EN15907:

- It makes a clear distinction between the levels of a digital resource: a work, a variant, a manifestation and an item (like FRBR).
- Resource based (events and agents) → Context information.

Disadvantages of EN15907:

- High level of granularity does not make it always suitable for mapping.

This standard is particularly suitable for the description of contemporary (media) art. Contemporary (media) art typically also makes a distinction between the artwork, its manifestations and related context metadata.

The European standard EN15744 is a metadata set for the identification of cinematographic work. It defines a set of data elements that are relevant for the identification of audio-visual creations at a work level. It can be considered as a core set of the EN15907 standard. Although it describes an audio-visual creation it includes some properties of an incarnation of the work in an audio-visual medium.

3.5 Mets, OAI-ORE and LIDO for exchanging Cultural Heritage Information

In this section, we introduce some common metadata harvesting models. These models can be used for exchanging metadata between a data provider and a data consumer, e.g., metadata harvesters. However, this does not imply that the data provider should use such models internally as the data model for storing their information. Not all of the proposed models are suitable for internal use. The models introduced are METS (3), OAI-ORE (28), and LIDO (18). These three models were chosen because at the moment they are popular for exchange purposes within the cultural heritage context.

3.5.1 METS

The Metadata Encoding and Transmission Standard, METS, is a specification for describing and exchanging digital objects and their information. METS is an open standard which was originally designed by the library community, but which is also being more and more used by archiving institutions. One of the advantages of METS is that it can hold the binary streams of a multimedia asset. This model packages the metadata together with their reference multimedia objects. This property of METS makes it popular for use in archives, where metadata and multimedia content have to stay together in defined packages, and for metadata exchange purposes because whole multimedia assets can also be exchanged at the same time.

METS has a hierarchical structure and is, as such, able to express the hierarchy of digital objects. For this, METS is structured out of different “METS elements”. Each METS element describes a digital object and can be related to other METS elements. Each element offers some sections, which contribute to both the management of the data and its exchange. Each section covers a certain type of metadata (descriptive, structural, …) Such a metadata model could therefore also be used internally as a data model for an institution's platform.

The sections of a METS element are:

```
<mets>
        <dmdSec/>
        <amdSec/>
        <fileSec/>
        <structMap/>
        <structLink/>
        <behaviorSec/>
</mets>
```

The dmdSec section covers the descriptive metadata section. It acts as a wrapper, meaning that one could include elements of other metadata models in the wrapper. Such wrappers make METS very modular and extensible. For the content of the wrapper, METS does not foresee any vocabulary or syntax. That is the responsibility of the included metadata model. In practice, there already exist some extensions to METS, including PREMIS or MARC XML. The data of these wrappers does not need to be textual. It can also be binary, as with e.g., MARC21.

The amdSec section covers administrative metadata needed for the management of the digital object. This section is actually also a wrapper, in which elements of other metadata models can be included.

The fileSec section offers information on the files that belong to the digital object. These files can be included into the metadata model. As mentioned earlier, one can also include non-textual information in METS, i.e., a binary representation of a file could be included in the metadata itself.

The files can also be referenced in this section, including a link to the file, instead of the file itself.

The section structMap gives information on the structure of the digital object. This allows for the description of the hierarchical structures of digital objects and how they relate to each other. This section creates the possibility to store several structures for each digital object. In such a way one can describe the logical structure of a digital object as well as the physical structure of the object. The structLink section of METS will present the hyperlinks between the several components of the structMap section.

The last section, the behaviorSec section, is used to connect several digital objects to applications or programme implementations, which can be used in combination with other information included in METS to render the digital objects.

Advantages of Mets:

- Extensible
- Modular
- Can hold bit streams

Disadvantages of Mets:

- Possible security issues, when including programme codes in the behaviorSec section.
- Not for layman.

3.5.2 OAI-ORE

Today, many information systems, e.g., content management systems, support the storage and identification of aggregations, as well as access to aggregations and aggregated objects. In most systems these objects vary in semantic type (e.g., article, book, video, dataset, etc.), and in metadata file format (e.g., PDF, XML, MP3, etc.). These objects can also be stored on different network locations, i.e., aggregated objects can be stored locally or externally. Information systems store, identify, and deliver access to these compound objects in an architecture-specific manner. Unfortunately, a lot of the advanced functionalities get lost when publishing the compound objects onto the Web.

The OAI-ORE standard has developed a standardised, interoperable and machine-readable mechanism that can express the information of compound objects. The standard makes sure that the logical boundaries of the aggregated objects and their mutual relations remain intact for machine agents when publishing the compound object onto the Web. OAI-ORE focuses on describing these aggregations and the relations between the aggregated resources. It does not only provide a metadata model for describing aggregations and aggregated resources, but also defines a protocol on how to automatically discover the metadata of an aggregation. The OAI-ORE metadata model is often used for exchanging metadata between a provider and consumer. That is why we focus primarily on the metadata model behind the OAI-ORE specification in this chapter.

OAI-ORE offers a metadata model describing resource maps and a protocol to discover resource maps automatically. These resource maps are RDF (machine-readable) descriptions of the aggregation. They list the aggregated resources, their mutual relations and the web context of the aggregation, together with the URI of the resource it is describing, i.e., the aggregation. Actually, these resource maps are named ‘graphs’. They are RDF graphs, sets of triples, extended with a name, a URI, for the graph/resource map. The named graph is not the aggregation itself, but a representation of its description encoded in Atom or RDF/XML, as depicted in the RDF/XML document below. The ORE model demands that a resource map describes just one aggregation. An aggregation, on the other hand, can have multiple resource maps, each with its own representation. This makes it possible to describe the same aggregation, for instance, with an RDF description and an XHTML description.

OAI-ORE specifies a metadata model for describing the resource maps that describe the resource maps.


```xml
<?xml version="1.0" encoding="utf-8"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:ore="http://www.openarchives.org/ore/terms/"
  xmlns:dcterms="http://purl.org/dc/terms/"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:foaf="http://xmlns.com/foaf/0.1/" >

  <!-- About the Aggregation for the Example document -->

  <rdf:Description rdf:about="http://example.org/aggregation/DCA/0601007">
    <!-- The Resource is an ORE Aggregation  -->
    <rdf:type rdf:resource="http://www.openarchives.org/ore/terms/Aggregation"/>
    <!-- The Aggregation aggregates ... -->
    <ore:aggregates rdf:resource="http://example.org/abs/DCA/0601007"/>
    <ore:aggregates rdf:resource="http://example.org/ps/DCA/0601007"/>
    <ore:aggregates rdf:resource="http://example.org/pdf/DCA/0601007"/>
    <!-- Metadata about the Aggregation: title and authors -->
    <dc:title>Example Aggregation that consists of a web page, an image and a document.</dc:title>
    <dcterms:creator rdf:parseType="Resource">
      <foaf:name>Sam Coppens</foaf:name>
      <foaf:mbox rdf:resource="mailto:samcoppens@example.be"/>
    </dcterms:creator>
    <dcterms:creator rdf:parseType="Resource">
      <foaf:name> Erik Mannens</foaf:name>
    </dcterms:creator>
  </rdf:Description>

  <!-- About the Resource Map (this RDF/XML document) that describes the Aggregation -->

  <rdf:Description rdf:about="http://example.org/rem/atom/DCA/0601007">
    <!-- The Resource is an ORE Resource Map  -->
    <rdf:type rdf:resource="http://www.openarchives.org/ore/terms/ResourceMap"/>
    <!-- The Resource Map describes a specific Aggregation   -->
    <ore:describes rdf:resource="http://example.org/aggregation/DCA/0601007"/>
    <!-- Metadata about the Resource Map: datetimes, rights, and author -->
    <dcterms:modified>2008-10-03T07:30:34Z</dcterms:modified>
    <dcterms:created>2008-10-01T18:30:02Z</dcterms:created>
    <dc:rights>This Resource Map is available under the Creative Commons Attribution-Noncommercial Generic
license</dc:rights>
    <dcterms:rights rdf:resource="http://creativecommons.org/licenses/by-nc/2.5/rdf"/>
    <dcterms:creator rdf:parseType="Resource">
      <foaf:page rdf:resource="http://example.org"/>
      <foaf:name>example Repository</foaf:name>
    </dcterms:creator>
```

```
  </rdf:Description>

  <!-- About the human start page that is part of the Aggregation -->

  <rdf:Description rdf:about="http://example.org/abs/DCA/0601007">
    <dc:format>text/html</dc:format>
    <dc:title>the web page</dc:title>
  </rdf:Description>

  <!-- About the PostScript resource that is part of the Aggregation -->

  <rdf:Description rdf:about="http://example.org/ps/DCA/0601007">
    <dc:format>application/postscript</dc:format>
    <dc:language>en</dc:language>
    <dc:title>the image </dc:title>
  </rdf:Description>

  <!-- About the PDF resource that is part of the Aggregation -->

  <rdf:Description rdf:about="http://example.org/pdf/DCA/0601007">
    <dc:format>application/pdf</dc:format>
    <dc:language>en</dc:language>
    <dc:title>and the document</dc:title>
  </rdf:Description>
</rdf:RDF>
```

Finally, we briefly describe the discovery protocol of OAI-ORE, which is less relevant for harvesting purposes. Clients and applications need to determine the URI of the resource map from the URI of the aggregation to get a description of the aggregation. This can come about in two ways. One is to append a fragment identifier ("#") to the URI of the resource map. For instance, the URI "http://example.com/aggregation" is the URI of the resource map, "http://example.com/aggregation#" is the URI of the aggregation. If a web crawler lands on the URI of the aggregation, it can strip off the fragment identifier to get a description of the aggregation. Cool URIs offer another solution, e.g., by appending the 'RDF' extension to the URI of that aggregation to obtain the URI of the resource map describing that aggregation.

In practice, this means that every aggregation should get a URI, like any resource on the Web. From this URI, a web agent should automatically be able to get a machine-readable description of the aggregation, namely the resource map. This resource map also has, of course, a URI. This URI should be deducted from the URI of the aggregation. This is done by, for instance, using 'cool URI'. The web agent adds 'RDF' to the URI of the aggregation and gets its machine-readable description.

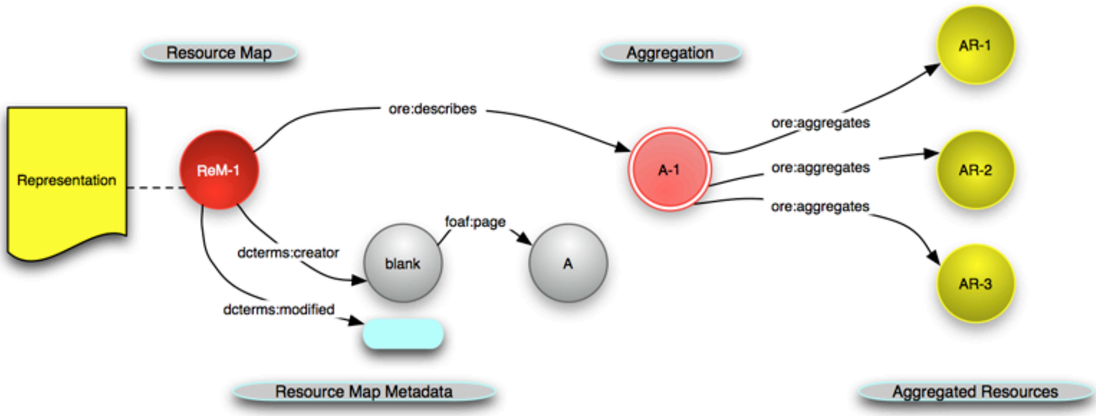


Figure 8: Schematic overview of OAI-ORE

Advantages of OAI-ORE:

- Describes aggregations
- Accessible for web crawlers
- Allows hierarchy
- Already has a binding with RDF

Disadvantages of OAI-ORE:

- Protocol has to support the OAI-ORE model

3.5.3 LIDO

LIDO is a pure harvesting metadata model. It is intended to deliver metadata on museum objects in a service environment, like portals, harvesters, online collection databases, etc. The strength of this model is that it can handle many different descriptive metadata models specific to the museum domain. LIDO is based on CDWA Lite, CIDOC-CRM, museumdat, and SPECTRUM. LIDO itself is quite a large data model, allowing fine-grained descriptions of museum objects. At the same time, the LIDO data model requires only a limited number of fields. This means that institutions can decide which data they want to provide and publish online.

LIDO, like METS, consists of a set of nested 'wrappers', which structure the descriptions in relevant blocks to describe museum objects. An important part of this specification is the 'events' wrapper. This wrapper originated from the CIDOC-CRM specification and is very specific to museum objects. It is able to describe the creation of the object, its transfer to a collection, the use of the object, etc. These are all described as events.

Conceptually, there are seven 'wrappers' in a LIDO record. Four of them have a descriptive character and three of them have an administrative character. The four descriptive wrappers are:

- Object Identification elements: This wrapper stores the elementary data of the object, like title, description, the repository it comes from, its dimensions, some display and edition information, and inscriptions, which are transcripts and/or its description.
- Object Classification elements: This wrapper gives information on the type of the object. The object name and other classification terms, like form, sex, age, state, etc., can be stored in this wrapper.
- Relation elements: In this part relations of the object to other resources are described. These

other resources can be concepts from a vocabulary, events, locations, other objects, dates and agents (or actors).

- Event elements: This section is responsible for describing events that are relevant for the object. Events can be the creation of the object, its transformation, publication or exhibition, etc. This events wrapper can also be used to store the provenance of a record.

The three administrative wrappers are:

- Rights for Work elements: This is for rights metadata. It describes the rights associated with the object, its metadata and the digital version of the object (surrogate). Here, we can find information on rights holders, rights type, dates and credit lines.
- Record elements: This is for some administrative data, like the ID, the type, the source, record rights (rights on the metadata, because these can be different from the rights on the object), and metadata references.
- Resource elements: This wrapper gives information about the digital resource being supplied to the service environment. The information stored in this wrapper are the URL of the resource, the resource ID, its relationship type, its resource type, the resource rights (because they can be different from the object rights), view descriptions, view type, view date, resource source (if not from the holding organisation), related resources and resource metadata location (pointer to other information about the resource).

Advantages of LIDO:

- Specially designed for exchanging cultural heritage information.
- Targeted for import in Europeana.
- Distinction between artwork, its representations and its relating documents.

Disadvantages of LIDO:

- Application profiles still in development.

4 Data Value Standards

In this section some aforementioned vocabularies will be discussed in detail. Each vocabulary has its own categorisation, its own purpose and its own multilingual support. The vocabularies that will be discussed are:

- AAT
- TGN
- ULAN
- RKDArtists

4.1 AAT

The Art and Architecture Thesaurus (AAT, (22)) consists of 34,000 concepts, related to art and architecture. The concepts cover a time span from antiquity until now. A unique numerical ID identified each concept. Each concept is related to terms, related concepts, a parent (place in the hierarchy), sources of data and notes.

This way the thesaurus knows 131,000 terms, related to 34,000 concepts. These terms are used to describe art and architecture. The terms of a concept contain the singular form, the plural form, the natural order, spelling variant, possible pronunciations, and synonyms. One of the terms related to a concept is indicated as the preferred term or the descriptor.

The AAT is a hierarchical database. It is designed to provide a classification of art and architecture. The facets within AAT form the most important classification of the concepts. A facet contains a homogeneous class of concepts of which terms contain characteristics that distinguishes them from other facets. For instance, marble belongs to the facet of materials used in art and architecture. The facets are organised hierarchically, from abstract to concrete, physical artefacts.

The main facets are:

- Associated Concepts: This facet contains abstract concepts and phenomena related to art and architecture, e.g., beauty, balance, freedom, etc.
- Physical Attributes: This category provides perceptual or measurable characteristics of materials and artefacts, e.g., round, brittle, boundaries, ...
- Styles and Periods: This facet contains common terms referring to stylistic tendencies and periods relevant for art and architecture, e.g., Louis XIV, Abstract Expressionism, ...
- Agents: This category describes terms denoting people, organisations that are identified by their occupation or activity, physical or mental characteristics, social role, etc. Examples are corporations, religious groups, architects, etc.
- Activities: In this group we find terms related to an activity in general. Examples are archaeology, designing, analysis, ...
- Materials: This category describes a physical substance, e.g., iron, clay, etc.
- Objects: This group contains terms for objects produced by humans, e.g., paintings, cathedrals, gardens, ...

The main advantage of this vocabulary in the context of the DCA project is its breadth of capacity to describe art works. The main disadvantage is that it might lack some specific terms related to contemporary art.

4.2 TGN

The Getty Thesaurus of Geographic Names (23) contains 912,000 records on places, from prehistory until now. The structure is very similar to that used by the AAT. Every record refers to a place and is related to a unique numerical ID, names, a parent, geographical coordinates, notes, sources for data, a place type and other relations. The place type describes the role of the place, e.g., capital of a state. The thesaurus contains in total 1,106,000 names. The names are expressed in English but also in the local language and sometimes in other languages.

Just like the AAT thesaurus, the TGN thesaurus is categorised hierarchically. At the root, there are two main facets: World and Extra-terrestrial Places. The places are divided into categories in their current political and physical world, e.g., World – Europe – United Kingdom – England.

The main advantage of this vocabulary is its inclusion of geographical coordinates and categorisation to disambiguate between places with the same name, as well as its multilingual support. A disadvantage is that it may not always include places relevant for contemporary art.

4.3 ULAN

The Union List of Artist Names (ULAN, (24)) is a thesaurus for artists. It contains the names of 120,000 artists. The structure is again very similar to that of thesauri previously discussed, i.e., TGN and AAT. A record denotes an artist. Each record is related to a unique numerical ID, names, related artists, sources of data and notes. The thesaurus contains artists from antiquity until now. It contains about 293,000 names of artists, including spelling variations, pseudonyms, names in different languages, ... The ULAN thesaurus has little hierarchy. The root has only two subdivisions: Person and Corporate Body.

The advantage of using this vocabulary is the inclusion of different spelling variations of the names, which benefits the retrieval of the data. A main disadvantage in the context of the DCA project is that it may not always contain names of contemporary artists who are internationally less known.

4.4 RKDArtists

RKDArtists (26) is a database with bibliographic data on Dutch and foreign artists from the Middle Ages until now. The database also provides information on art traders, art collectors and art historians. The documentation on an artist includes images, archives, press documentation, and literature. Cultural heritage institutions often use this database as a thesaurus for the dissemination of their content. The database currently consists of about 250,000 records. Each record has the following properties:

- a record ID
- a preferred name
- name variants
- sex
- date of birth and death
- place of birth and death
- periods active as artist
- sources of data
- literature and documentation.

This vocabulary has the same advantages and disadvantages as the ULAN vocabulary. It might be more relevant for Dutch (and Flemish) artists, while ULAN is more internationally oriented.

5 Guidelines for Metadata Exchange of Contemporary Art

In this chapter, guidelines are provided for exporting metadata on contemporary art. Within the framework of the DCA project the first destination for contemporary art is Europeana. For this, we investigate the metadata models that Europeana is using or will be using. Then we have a look at another possible aggregator targeted specifically to media art, a subset of contemporary art, i.e., GAMA. Currently these are the main candidates for ingesting DCA contemporary art data. Based on the researched data models, we recommend a metadata model for exchanging contemporary art metadata. This model must be able to retain the different levels of a digital contemporary art resource and to map as lossless as possible the discussed metadata models. As such, we would like to show that disseminating is possible with this recommended exchange metadata model to aggregators like Europeana, but also to other aggregators that want to harvest metadata on art.

5.1 Metadata for Europeana

In this section, we will elaborate on the metadata that Europeana is expecting. Europeana has two data models at the moment: Europeana Semantic Elements (ESE, (29)) and the Europeana Data Model (EDM, (30)). The ESE model was the first data model in use in Europeana. It is very concise and based on the Dublin Core schema. This model, however, does not allow for specialised descriptions of the objects. It is very flat and generic, but too concise. Museums, for instance, typically want to be able to describe aggregations of objects, their relationships, the events relating to that object, e.g., like exhibitions and loans to another museum, etc. The EDM model covers many of these disadvantages. These two models are described in the next two subsections.

5.1.1 Europeana Semantic Elements (ESE)

This schema is the original metadata model used by Europeana to describe all harvested records on a common basis. It is in fact an application profile of the Dublin Core model that can be used to describe a heterogeneous set of records.

This ESE model consists of a set of terms originally from the Dublin Core and Dublin Core Terms. Europeana then added some specific ‘Europeana’ fields, targeting its specific needs. The terms of the ESE model can be divided into four categories: highly recommended, recommended, additional elements and Europeana elements. All terms listed in the ‘highly recommended’ category are essential for dissemination to Europeana. They ensure that the record can be retrieved from the Europeana repository. The Europeana framework indexes such elements. The terms of the ‘recommended’ category are also very important. Together with those of the previous category, they are able to answer queries about ‘What’, ‘When’, ‘How’ and ‘Who’. They are also, therefore, needed for retrieval purposes. Table 1 shows the terms of the ESE model, divided into the four categories.

Highly recommended	Recommended	Additional elements	Europeana elements
dc:title	dc:coverage	dc:format	europeana:country
dcterms:alternative	dcterms:spatial	dcterms:extent	europeana:hasObject
dc:creator	dcterms:temporal	dcterms:medium	europeana:isShownAt
dc:contributor	dc:description	dc:identifier	europeana:isShownBy
dc:date	dcterms:isPartOf	dc:rights	europeana:language
dcterms:created	dc:language	dcterms:provenance	europeana:object
dcterms:issued	dc:publisher	dc:relation	europeana:provider
	dc:source	dcterms:conformsTo	europeana:type
	dc:subject	dcterms:hasFormat	europeana:unstored
	dc:type	dcterms:isFormatOf	europeana:uri
		dcterms:hasVersion	europeana:usertag
		dcterms:isVersionOf	europeana:year
		dcterms:hasPart	
		dcterms:isReferencedBy	
		dcterms:references	
		dcterms:isReplacedBy	
		dcterms:replaces	
		dcterms:isRequiredBy	
		dcterms:requires	
		dcterms:tableOfContents	

Table 1: ESE element Classification

Europeana also has some rules, which will guide the data provider in mapping their metadata to the Europeana ESE model:

- Try to map as many fields as possible to the ESE model.
- When it is not possible to map a certain field to an element of the ESE model, leave the field unmapped or use the ESE element Europeana:unsorted.
- If it is possible to choose between several ESE elements to map to, choose the most specific one. For instance, dc:converage denotes a place or period in time, but when you are sure about denoting a location, choose the dcterms element dcterms:spatial.
- The persistent link to the digital object and/or the information page must be dereferencable URLs, i.e., you should be able to follow the links in a browser.
- When you are not sure which ESE element to map to, think about the end-user and choose the element that best reflects the end-user's needs.
- When there are several values for the same ESE element, repeat the ESE element for every value.
- Include units whenever possible, e.g., 100 x 100 pixels.

Concerning the specific Europeana elements, only europeana:isShownAt, europeana:isShownBy, europeana:object, europeana:provider, europeana:type, and europeana:unsorted must be delivered by the data provider. All the other ‘Europeana specific’ terms will be filled in by the Europeana platform itself. This does not imply that these are the only mandatory fields in ESE, only those of the Europeana specific terms.

5.1.2 Europeana Data Model (EDM)

The ESE model of Europeana is very concise. Anticipating the new semantic release of the Europeana platform, the Danube release, this ESE model is too restricting. That is why a new one was developed that will be used in the Danube release. For the time being, EDM is only in use within a Europeana LOD pilot.

With this metadata model, Europeana addresses the criticism of being focused too much on libraries. The museum sector in particular had a lot of trouble mapping the ESE model. The ESE model is not resource-based, i.e., one only has one sort of resource, i.e., the Europeana object. This forms a major obstacle when enriching and interlinking the data to external resources. One is only able to link to other Europeana objects, but not to resources that describe, e.g., locations or persons.

The EDM model is resource-based, meaning it consists of many different types of resources that are linked to each other. The types of resources in the EDM model are: Europeana aggregations, events, agents, locations, physical objects, concepts and time spans. With the EDM model, one is able to relate all the different types of resources to each other to form a very precise description of the object and its manifestation. Every resource can also be linked to external resources, which provides one with additional information on that resource. In the end, the EDM model allows one to describe data more precisely and to interlink it on a more granular level. The scheme below gives an overview of the EDM model.

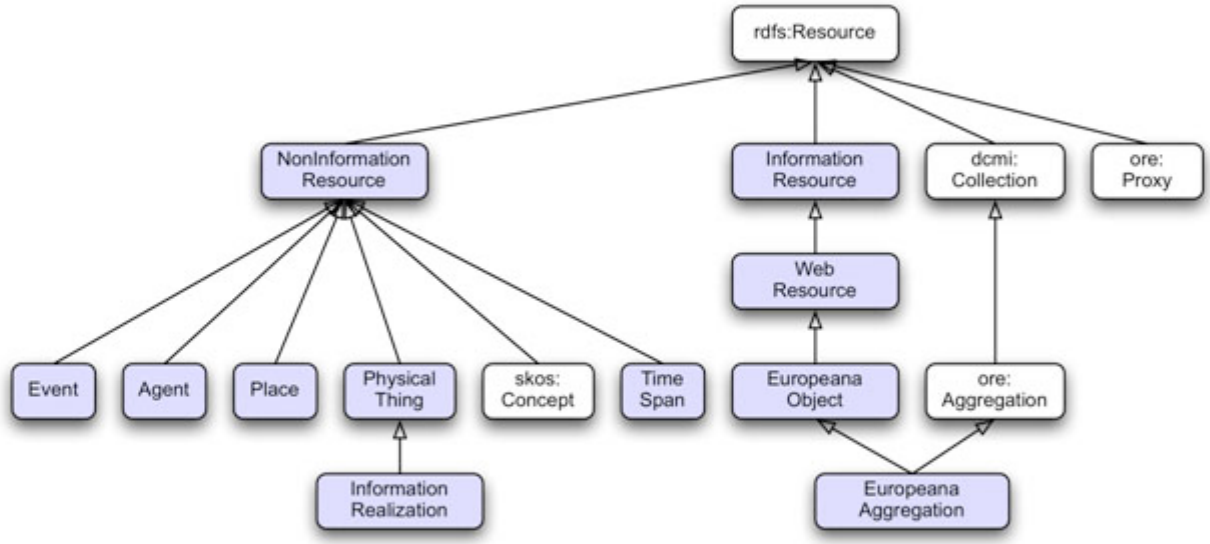


Figure 9: Europeana Data Model

For describing some of the resources, Europeana reuses some existing ontologies. Others are described within the EDM model. Below, the resources used in EDM are briefly explained:

The resources described reusing ontologies:

- FOAF Agent: This class describes agents. Agents can be persons, organisations, or groups. Sometimes a piece of software is also described as an agent.
- IRW Resource: This class describes resources in general. It corresponds to the RDFS resource. Every resource in EDM is by default an IRW Resource.
- IRW Information Resource: information resources are those that can describe nearly everything, like, e.g., analogue resources. When a resource has a URI, it becomes an IRW

WebResource. An information resource can also have a realisation, e.g., Moby Dick's text is an information resource; the book of that text is a realisation/manifestation of the information resource, i.e., the text.

- IRW Non-Information Resource: All semantic resources that are not resources of information are called non-information resources. It includes persons, time spans, locations, physical objects, etc.
- IRW Web Resource: These are information resources that have a URI and can be looked up on the Web.
- ORE Aggregation: These resources are aggregations, i.e., sets of related resources. They enable the description, e.g., of collections or compound objects like, e.g., books, which can be an aggregation of its pages, and are described as separate resources.
- ORE Proxy: A proxy is a resource that describes an aggregated resource of an aggregation from the point of view of that proxy. A collection can contain more resources in the context of a certain institution, than in the context of another. These differences can be described using ORE:Proxy.
- SKOS Concept: a SKOS concept describes a concept/term of a vocabulary, which can be related in a hierarchical manner.

Specific EDM Resources are:

- Europeana Aggregation: These are sets of resources that are related to one physical cultural heritage object. Europeana tries to aggregate all records that describe the same object.
- Europeana Object: any digital object of which Europeana is the rights holder.
- Event: describes events relevant to the Europeana object.
- Physical Thing: any persistent physical object, like, e.g., a book, a painting, etc.
- Place: locations, more specifically ‘locations on the surface of the Earth’
- Time Span: describe time spans.

Except for the aforementioned, reused resources, EDM determines which properties are used to describe the specific EDM resources or to link it to others. The properties are depicted in the figure below.

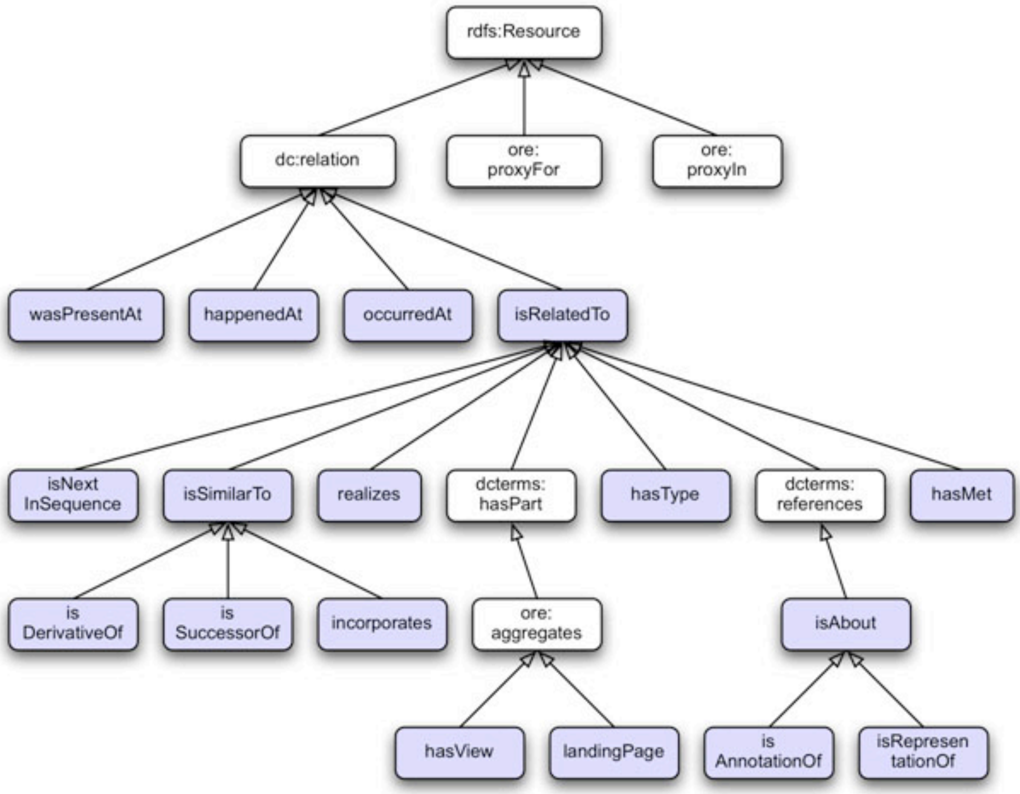


Figure 10: EDM properties

It is clear that Europeana aims to publish the metadata as Linked Open Data with this EDM model. It also addresses the shortcomings of the ESE model, in particular for the museum sector.

5.2 Metadata for Media Art

Lastly, we introduce a metadata model that was designed by the media art community in the GAMA project. It is a domain-specific metadata model that reflects the specific needs of the media art sector.

5.2.1 GAMA

GAMA – Gateway to Archives of Media Art (http://www.gama-gateway.eu/) was launched in 2007 by 19 participating organisations from Europe's media art sector. The European commission supported this project under the eContentPlus programme. The goal of the project was to create a central portal site, giving access to several media art collections. The end-users of this portal site are curators, interested audiences, researchers and mediators.

The GAMA portal provides multilingual access to various media art archives and their digitised content. The content that the portal publishes covers 55% of the European media art that is currently accessible online.

The metadata model that GAMA uses to unify the media art content is described using RDF. The namespace of the metadata schema is http://gama-gateway.eu/schema/ or gama in short. It consists of 11 classes, which can be subdivided into two groups: entities and enumerations.

The entities are described using the following classes:

- gama:Work — Artworks, events and other data sources
- gama:Person — Persons, institutes, collectives
- gama:Manifestation — Physical representations of artworks
- gama:Archive — Archives
- gama:Collection — Collections of artworks

The enumerations are classes with fixed instances. They constitute the following classes:

- gama:WorkType — List of types of artworks
- gama:MediaType — List of media types
- gama:Genre — Hierarchical list of genres
- gama:Term — List of frequently used terms to describe media artworks
- gama:Country — List of countries
- gama:Language — List of languages for describing the content

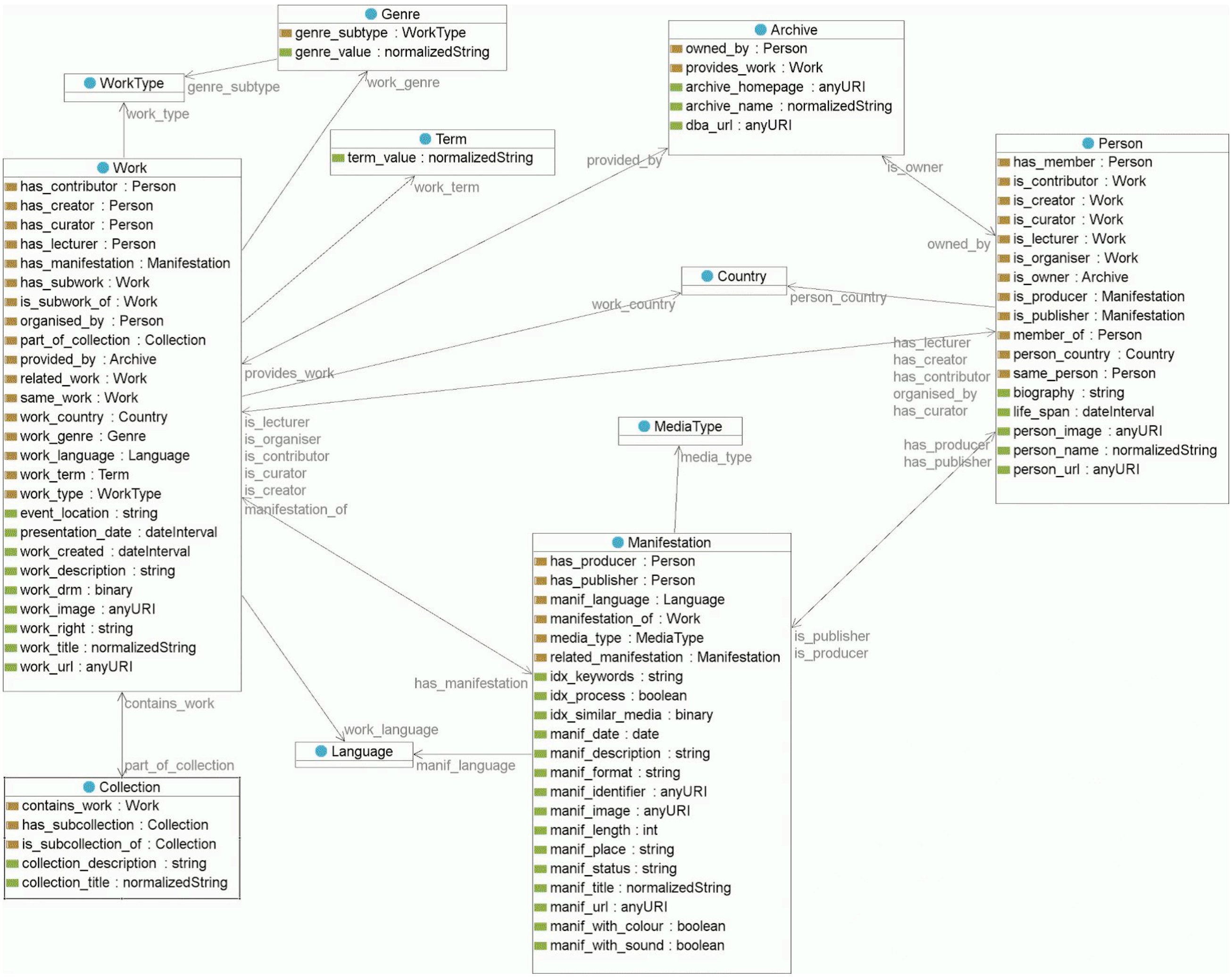


Figure 10: Overview of the GAMA model

5.3 Exchange Metadata for Contemporary Art

The previous sections gave an overview of:

- Some harvesting models that can be used for aggregating and disseminating contemporary art content (LIDO, METS, Dublin Core, etc.).
- The Europeana data models that are in use (ESE) or under consideration (EDM).
- A specific metadata model, i.e., GAMA, designed by the media art community and for the media art community to be used for aggregating media art information from different data providers.

From these models, we filter the necessary terms that content providers of contemporary art have to disseminate in order to obtain a meaningful dissemination through Europeana. As a starting point, the LIDO terms are evaluated. Next we extend this preliminary set of terms with those that are specific for contemporary art. The GAMA model serves as a basis for such terms. As such, we combine the best of both models.

There are three reasons why LIDO was chosen as a basis for selecting the necessary terms for describing contemporary art that will be harvested by Europeana. First, the LIDO data model is a good harvesting model that is able to describe records from different domains, e.g., libraries, museums or archives. The second is that it can be easily used for ingestion into Europeana. Projects like ATHENA have already proved this with their LIDO mapping and ingestion tool. Thirdly, the LIDO model is able to retain the different levels of a digital contemporary art resource.

In this section, we start with the minimum requirements of a LIDO model. Which fields have to be provided in order to get a valid LIDO model? Such description can thus be provided to Europeana for dissemination. But this is not good practice. Europeana will be able to publish the LIDO record, but the description of the object is too concise to become retrievable in Europeana.

In order to obtain a good description of the contemporary art object and to enhance discoverability of these descriptions, we propose an application profile of LIDO. It is in fact a LIDO model, but targeted towards descriptions of contemporary art objects. We will call it the DCA application profile of LIDO for the remainder of the document. In fact, a similar exercise is currently being made in the Linked Heritage community. They are producing profiles of LIDO targeted towards certain domains. One of the application profiles they will produce is a LIDO profile for describing fine art. Eventually, this fine art profile of LIDO will become the recommended metadata model for the exchange of contemporary art metadata. Meanwhile, we are producing our own profile of LIDO, i.e., the DCA application profile of LIDO.

This DCA application profile is in fact a combination of the LIDO model and other models that are suited for describing contemporary art, e.g., the GAMA model or the EN 15907 standard. The GAMA model will be mapped to the LIDO model in order to obtain the DCA application profile. The EN 15907 and LIDO are not explicitly mapped to each other, but are actually very similar in structure and both retain the different levels of a digital contemporary art resource. They map one-on-one.

Of course, we will also introduce some best practices, e.g., on resource-based work and on introducing vocabularies. Based on this application profile, we will then be able to filter out all necessary metadata fields that a contemporary art institution needs to provide in order to be 'mappable' to this DCA application profile of LIDO. The vocabularies introduced will be discussed in the next section.

Such a strategy will ensure that our DCA application profile of LIDO is mappable to the ESE or EDM model of Europeana, but also to the GAMA model, which could be another data consumer of the contemporary art metadata. This application profile of LIDO is not only meant for dissemination to Europeana. Other aggregators could also benefit from it, which is why we have decided to map the DCA application profile of LIDO to the GAMA model. Our more developed DCA application profile of LIDO can therefore also be used beyond the context of Europeana to support other aggregation efforts, e.g., GAMA.

The taxonomy below gives an overview of the LIDO metadata model with all its tags:

- Wrapper for the whole document (lidoWrap)
- Wrapper for an object record (lido)
    - LIDO Metadata Record-ID (lidoRecID)
    - Published Object Identifier (objectPublishedID)
    - Category (category)
        - Concept Identifier (conceptID)
        - Term / Label (term)
- Descriptive Metadata (descriptiveMetadata)
    - Object Classification Wrapper (objectClassificationWrap)
    - Object Identification Wrapper (objectIdentificationWrap)
    - Event Wrapper (eventWrap)
    - Object Relation Wrapper (objectRelationWrap)
- Administrative Metadata (administrativeMetadata)
    - Rights for Work Wrapper (rightsWorkWrap)
    - Record Wrapper (recordWrap)
    - Resource Wrapper (resourceWrap)

5.3.1 Minimum Requirements

The LIDO metadata model stipulates that a minimum of the following elements must be present in a LIDO model:

- **lidoWrap** (NR; ; lido)
- **lido** (R; **lidoWrap**; lidoRecID, objectPublishedID, category, descriptiveMetadata, administrativeMetadata)
    - **lidoRecID** (R; **lido**;)
    - **administrativeMetadata** (R; **lido**; rightsWorkWrap, recordWrap, resourceWrap)
        - **recordWrap** (NR; administrativeMetadata; recordID, recordType, recordSource, recordRights, recordInfoSet)
            - **recordID** (R; **recordWrap**;)
            - **recordSource** (R; **recordWrap**; legalBodyID, legalBodyName, legalBodyWeblink)
            - **recordType** (NR; **recordWrap**; conceptID, term)

    - **descriptiveMetadata** (R; **lido**; objectClassificationWrap, objectIdentificationWrap, eventWrap, objectRelationWrap)
        - **objectClassificationWrap** (NR; **descriptiveMetadata**; objectWorkType, classificationWrap)
            - **objectWorkTypeWrap** (NR; objectClassificationWrap; objectWorkType)
                - **objectWorkType** (R; objectWorkTypeWrap; conceptID, term)

- **objectIdentificationWrap** (NR; **descriptiveMetadata**; titleWrap, inscriptionsWrap, repositoryWrap, displayStateEditionWrap, objectDescriptionWrap, objectMeasurementWrap)
  - **titleWrap** (NR; objectIdentificationWrap; titleSet)
    - **titleSet** (R; **titleWrap**; appellationValue, sourceAppellation)
      - appellationValue (R; eventName, legalBodyName, nameActorSet, namePlaceSet, **titleSet**;)

- **linkResource** (NR; resourceRepresentation;)
- **measurementType** (R; measurementsSet, resourceMeasurementsSet;)
- **measurementUnit** (R; measurementsSet, resourceMeasurementsSet;)
- **measurementValue** (NR; measurementsSet, resourceMeasurementsSet;)
- **nameActorSet** (R; actor; appellationValue, sourceAppellation)
- **eventType** (NR; event; conceptID, term)

These properties are the minimum requirements for a LIDO description of an object. First let's analyse this list. The root element of a LIDO document is lidoWrap. This is in fact a wrapper that can support many LIDO elements. Each LIDO element supports an object description.

A LIDO object description must have the following elements:
- ***lidoRecID***: An ID for the lido object record. This is repeatable, so a ***lido*** object record can have multiple IDs.
- ***administrativeMetadata***: A wrapper holding administrative metadata about the object record. It is repeatable.
- ***descriptiveMetadata***: A wrapper containing descriptive metadata about the object. This wrapper is also repeatable.

The ***administrativeMetadata*** wrapper supports the ***recordWrap*** element (non-repeatable), which gives information on the catalogue record itself. It must have at least the following elements:
- ***recordID***: an ID for the catalogue record (repeatable)
- ***recordSource***: reveals the source of information in this record, generally the repository or some other institution (repeatable)
- ***recordType***: provides information on the type of the record. A record can represent an individual item or a collection, series, or group of works. This is preferably taken from a controlled vocabulary.

The ***descriptiveMetadata*** field holds the descriptive information of the object record. It must have the following elements:
- ***objectClassificationWrap***: This wrapper holds classification information about the object/work (It is non repeatable): style, genre, form, age, sex, and phase or how the holding organisation structures its collection (e.g., fine art, decorative art, prints and drawings, natural science, etc.). This wrapper must at least support the ***objectWorkTypeWrap*** element. It is a non-repeatable element, which must hold at least one ***objectWorkType*** element, which is repeatable. This value is preferably taken from a controlled vocabulary.
- ***objectIdentificationWrap***: This wrapper holds information that identifies the object. It is non-repeatable. It must at least hold a ***titleWrap*** element. It includes the object name or title information, and is non-repeatable, but can hold several ***titleSet*** elements: at least one must be provided. This ***titleSet*** element holds one of the object's titles and consists of at least the ***appelationValue*** element. This repeatable element holds a string that forms the object's title.

All such elements make up a minimum description of an object. There are some other required elements, but only if their super element (the wrapper holding this element) is present. If the object description includes some actors (persons, organisations, groups) they must have at least a ***nameActorSet***. This element is repeatable and must have at least one ***appelationValue*** that denotes the actor's name.

The same holds true for the ***eventType*** element. Whenever an ***event*** is present in the object description, it must have at least an ***eventType*** element the value of which is preferably taken from a controlled vocabulary. This information is stored under the ***descriptiveMetadata*** element, which includes the ***eventWrap*** element. This ***eventWrap*** element can support several ***eventSet*** elements, which must all have at least an ***event*** element.

Until now, with all previous elements, we have been describing the object itself. Each object can have a resource. This is a digital representation of the object. Since DCA is a digitisation project, every object will indeed have a digital representation. Information about the resources is stored in the ***administrativeMetadata*** element. This element can include the ***resourceWrap*** element, although it is not necessary. This element is a wrapper for holding resources, which are surrogates for an object/work, including digital images, videos or audio files. This wrapper can support several ***resourceSet*** elements. Each ***resourceSet*** represents one digital representation about an object/work and must have at least one ***resourceRepresentation*** element. This field represents a digital representation of a resource. This element can be repeated for variants representing the same resource, e.g., different sizes of the same image. A ***resourceRepresentation*** must have at least a ***linkResource*** element, which includes a URI reference to the online representation of the resource.

Finally, one can describe the measurements of an object or its representation. Whenever storing information about the measurements (object or resource) one has to provide the following elements: ***measurementType***, ***measurementUnit*** and ***measurementValue***. When describing the measurements of an object, the information has to be stored in the ***measurementsSet*** element, which is part of the ***objectMeasurements*** filed. This field is part of the ***objectsMeasurementsSet***, belonging to the ***objectsMeasurementsWrap***. This wrapper is in turn part of the ***objectIdentificationWrap***. When storing measurement information about digital representation, the information has to be stored in the ***resourceMeasurementsSet*** wrapper, which is part of the ***resourceRepresentation*** element.

5.3.2 The DCA Application Profile of LIDO

DCA is a digitisation project, and as such, every described object will have a digital representation/resource, which will have to be described too. In order to obtain the DCA application profile of LIDO, we will map the GAMA model and the EN15907 to the LIDO model. This will result in a good LIDO description that is also suitable for media artworks and their digital surrogates.

The GAMA model consists of 11 classes that can be subdivided into two groups: entities and enumerations. In this section, we will focus on the first group, i.e., the entities. The enumerations of the GAMA model will provide a useful input for the controlled vocabularies to be used with the DCA application profile of LIDO (as will be discussed in the next section).

The entities are described in GAMA with the following classes:

- gama:Work Artworks, events and other data sources (lido: Object/Work)
- gama:Person Persons, institutes, collectives (lido: Actor)
- gama:Manifestation Physical representations of artworks, e.g., files or books (lido: Resource)
- gama:Archive Archives (lido: Repository)
- gama:Collection Collections of artworks (lido: Object/Work with objectWorkType denoted as a collection)

For every entity in GAMA, there is a corresponding element in the LIDO model. The corresponding LIDO elements are mentioned alongside the GAMA element description. From the GAMA model requirements, we can extract some for the lido model, which will become the DCA application profile of LIDO.

The main class of GAMA is ***Work***. This entity primarily represents artworks, but also events and other resources. The ***WorkType*** class further defines its type. LIDO makes an explicit distinction between artworks (lido objects/works), events and resources. In GAMA a ***Work*** can be an ***Artwork***, an ***Event*** or a ***Resource***. The LIDO object will always be a GAMA artwork. This means that GAMA artwork types can be used to denote the LIDO ***objectWorkType***.

An artwork in GAMA is defined by the following properties:

- Title: must be present in the ***titleSet*** as ***appellationValue.*** The titleSet element belongs to the LIDO titleWrap element. This titleWrap element is in turn part of the objectIdentificationWrap element.
- Year: dates in LIDO are a separate entity. A work can be related to a ***Date*** entity. This can be done in the LIDO ***subjectWrap***, which is part of the ***objectRelationWrap***. The ***subjectWrap*** has a ***subjectDate*** element. The year of the artwork can be stored in this wrapper. If the date represents the creation of the artwork, one can also create an event in LIDO and the date of this event can be filled in with the year. Such a method gives a more precise description.
- Type: defines the ***objectWorkType*** in LIDO. Its values should be taken from a controlled vocabulary. The list of GAMA ArtworkTypes is a possible candidate for this.
- Image: in LIDO images are digital surrogates, called resources. A reference to an image should be stored in the ***linkResource*** element of LIDO.
- Description: Descriptions of the artwork must be stored in the ***DescriptiveMetadata*** element of a LIDO artwork. This element must have an ***objectIdentificationWrap*** in which there is an ***inscriptionsWrap***. This wrapper can hold several ***inscriptions***, which can in turn have an ***inscriptionDescription*** or ***inscriptionTranscript*** (in case the description is a transcription). These are where GAMA descriptions are held.
- Country: A location is a separate entity in LIDO. To describe the country, one can follow the same workflow as for Year. The ***subjectWrap*** can include a ***subjectPlace***, where this value can be filled in. When the country represents the location of the creation of an artwork, one can make a creation event in LIDO and the country can be included in the ***eventPlace*** section.
- Collection: In LIDO an object can be a collection or an individual record. This is denoted in the ***category*** element of a lido object (***lido*** wrapper). Through the relations wrapper of LIDO one can related a record to its collection.
- Rights: The rights of the artwork itself must be stored in LIDO ***rightsWorkWrap*** element.

- Language: In LIDO the wrapper must have a language attribute for denoting the language of description. This means every wrapper can be repeated in any language for multilingual support. On the other hand, one can also use language attributes to any field in LIDO that has a String value. This way certain elements can just be repeated with a different language attribute for multilingual support, without having to repeat the whole wrapper.

The Person entity in GAMA corresponds to an Actor element in LIDO or actorInRole element (when an actor participates in an event with a certain role) and must have the following properties:

- Name: This must be filled in for LIDO as an ***appellationValue*** in the ***nameActorSet*** element under the ***actor*** element or ***actorInRole*** element.
- Role: If this element is present, the actor is participating in a certain event as a role. The ***Actor*** is then part of ***actorInRole***, which is related to an ***event***. The role can be described in the ***roleActor*** element.
- Country, Biography, and Year of Birth/Death: In order to store these data in LIDO, an actor has to participate in an event. One can then store extra information on the event such as location, dates and notes.
- Image: Images are surrogates in LIDO. An image of the Actor can be stored in the ***linkResource*** element of LIDO, which must be related to the ***actor***.

The Event entity in GAMA has a corresponding entity in LIDO. Its properties in GAMA are:

- Title: this value must be stored in LIDO as ***appellationValue*** under the ***eventName*** element of a LIDO ***event***.
- Year: this value must be stored in LIDO as ***date*** under the ***eventDate*** element of a LIDO ***event***.
- Type: every event must have an ***eventType*** in LIDO. This is stored in the ***eventType*** element of the ***event*** element in LIDO.
- Description: this is stored in the eventDescription element of LIDO that is part of the event element.
- Location: this information is included in ***eventPlace*** in LIDO.
- Organisation: an ***event*** in LIDO can be related to an ***actor***, which represents an organisation in LIDO.
- Language: as already mentioned in the description of the artwork class, any wrapper or field with a string value can be attributed with a language and can be repeated for multilingual descriptions.

The Resource entity in GAMA is defined by the following properties:

- Title: a resource in LIDO does not have a title. The object to which the resource belongs does have room for a title. Cfr. Supra.
- Year: a resource can have a date in LIDO. This is described by the ***date*** element. One also has a ***displayDate*** element for a resource description in LIDO. This element is a concise description of the date, presented in syntax, suitable for display by the end-user.
- Type: this defines the ***resourceType*** in LIDO. Its values should be taken from a controlled vocabulary. The list of GAMA MediaTypes is a possible candidate for this.
- Description: descriptions of the resource must be stored in the ***resourceDescription*** element of a LIDO Resource.
- Country: when the country represents the location of the creation of a surrogate, one can produce a creation event in LIDO and the country can be included in the ***eventPlace***

section.

- Collection: in LIDO an object can be a collection or an individual record. This is denoted in the ***category*** element of a LIDO object (***lido*** wrapper). Through the LIDO relations wrapper one can relate a record to its collection.
- Rights: the rights of the resource must be stored in LIDO ***rightsResource*** element, which has the same properties as the ***rightsWorkWrap*** wrapper.
- Language: in LIDO the wrapper must have a language attribute for denoting the language of description. This means every wrapper can be repeated in any language for multilingual support. On the other hand, one can also use language attributes for any field in LIDO that has a String value. This way certain elements can just be repeated with a different language attribute for multilingual support, without having to repeat the whole wrapper.
- URL: this gives the URL of the resource. A reference to a resource should be stored in the ***linkResource*** element in LIDO.

This section shows that LIDO and the GAMA model can be mapped to each other. The only restriction is that a resource in GAMA can have a title, whereas in LIDO doesn't. There is a slight difference in semantics for both elements though. The consequence of this mapping exercise is that we build a DCA application profile of LIDO based on the GAMA model.

From this mapping exercise we can now extract the fields that contemporary art institutions should provide for the description of their artworks and digitised surrogates in order to create a meaningful dissemination when the LIDO description is ingested into Europeana or some other harvesting platform.

We have made a distinction between the minimum required metadata fields. These are the fields that should be provided by content providers in order to obtain a minimum LIDO record. This is actually not good enough. Such minimum required fields do not support the retrieval of the content provided in the aggregator. To obtain good search possibilities, we supply the recommended fields. When the content providers submit these fields, the end user will be able to search the content and retrieve the contemporary art descriptions supplied. The additional fields will give extra context information that should be provided by the content partners, at least if they possess this metadata. The content partners should at least provide the minimum required fields as well as the recommended fields. Additional fields are optional, and will make the content more visible in the aggregator.

Every artwork might have at least the following properties:

Artwork	Minimum	Recommended	Additional
ID	X		
Title	X		
Date		X	
Type	X		
Description		X	
Place		X	
Measurements			X
Collection		X	
Rights		X	
Language		X	
Subjects/keywords	X		
Events			X
Surrogates			X

Every digitised item (surrogate) belonging to the artwork might have the following requirements:

Surrogate	Minimum	Recommended	Additional
URL	X		
Description		X	
Language			X
Date		X	
Type*		X	
Rights		X	
Measurements			X
Format (mimetype)		X	

If the content provider has information on events related to the artwork or resource, this information might also be provided and recorded into the LIDO description. An event might have the following properties:

Event	Minimum	Recommended	Additional
Title	X		
Date		X	
Type*	X		
Description		X	
Place			X
Actor			X
Language		X	

Every actor mentioned in an event might have the following properties:

Actor	Minimum	Recommended	Additional
Name	X		
Role*		X	
Place			X
Biography			X
Year of Birth		X	
Year of Death			X

6 DCA Vocabulary

We recommend using controlled vocabularies for some fields. This holds true for the artwork types, event types, surrogate types, keywords, and an actor's role. For dissemination to Europeana, we will provide a controlled vocabulary of keywords for denoting the type of artworks as well as the type of digital representation and any related documentation.

For the controlled vocabulary of keywords denoting the type of artworks, digital representations and related documenting resources, several strategies are possible. One could search for the greatest common divisor of all the keywords in use at the institutions, and employ this set of terms as a basis for the vocabulary. It must then be extensible in order to cover all terms (dynamic vocabulary). Another possibility could be to take the smallest common multiplier of all terms and use this as a definitive list for the vocabulary (static vocabulary). Finally, it could also be based on the keyword list used by the GAMA portal when it could be seen how all keywords might be mapped to it.

6.1 Analysis: Provided keywords vs. GAMA keyword

In this section, we shall examine the keywords used by all the data providers of the DCA project. The keywords used are displayed in the tag cloud below. The bigger the word, the more it is used by different contemporary art institutions. The words that are at least used by two institutions are shown in the table below.

Figure 11: Tag cloud of used keywords

Keywords
Video
Installation
Sculpture
Painting
Drawing
Collage
Poster
Relief
Video installation
Assemblage
Object
Film
Multimedia installation
Documentary
Engraving
Net art
Photography
Print
Book

Figure 12: Keywords used by more than one DCA content partner

The table therefore shows all keywords used by more than one institute. This list is actually not enough to cover all keywords or to map them to the selection provided, but it does cover a major part of them.

The GAMA project also has a vocabulary defined for artwork, event, and resource types. Except for the event types, this vocabulary can also be used. It has been created by the media art community and reflects many needs of contemporary art institutions. Media art is only a subclass of contemporary art, though, so it does not cover the whole spectrum of possible keywords. For instance paintings is an artwork type of contemporary art, but not of media art. The GAMA keywords are listed in the table below.

Artwork types	Resource types
Animation	Article
Computer graphics	Book
Documentary	Critique
Film	Document
Hybrid Art	Essay
Installation	Exhibition Item
Music	Festival Item
Net Art	Interview
Performance	Jury statement
Portrait	Programme
Sound Art	Radio
Television	Research Item

Video Art	Submission form
Web	Periodical
Artists' book	Multimedia
	Catalogue
	Press Text
	Report
	Thesis

Figure 12: GAMA Keywords

6.2 Recommendation

For the DCA keyword vocabulary, neither of the two aforementioned keyword lists will do the job. But they are complementary. As a starting point for the DCA vocabulary, we will merge the two aforementioned vocabularies into one, normalised vocabulary. The resulting vocabulary is only preliminary. It will be extended as the content partners map their content to the provided LIDO profile. A taskforce will act as editorial committee for the DCA vocabulary. It will decide if the vocabulary needs to be extended and under what terms. If a content partner cannot map a certain keyword to the DCA vocabulary, it is up to the taskforce either to suggest an alternative term that covers the semantics of the to-be-mapped keyword or to extend the DCA vocabulary. The result will be a vocabulary supported by all DCA content partners.

Initially the DCA vocabulary will be designed in English. Later, each term of the vocabulary will be translated into the language of the content partners. In the end, we will have a multilingual vocabulary for the keywords in DCA.

To support interoperability and to anticipate the use of the AAT vocabulary in Europeana, but also by other aggregators, we will map the DCA multilingual vocabulary to the terms of the AAT vocabulary. As such, we will obtain an interoperable, multilingual vocabulary for characterising contemporary art. This will greatly enhance the reuse of the vocabulary by content partners beyond the DCA project context and the Europeana aggregator context.

7 Metadata and Long-Term Preservation

In this chapter, we will briefly introduce the preservation of metadata. DCA is a digitisation project. A lot of digital material will be produced during this project. These digital resources should be preserved for the long-term. As content provider, one does not want to have to repeat the expensive exercise of digitisation. The digitised resources will need to be preserved long-term, i.e., they will need to stay intact and interpretable. This puts extra requirements onto metadata for contemporary art, so that it might support the necessary preservation processes. For this we require preservation metadata.

7.1 Risks of long-term archiving

When preserving digital information for the long-term, digital archives demand specific requirements. On the one hand, the software and hardware of the digital archive have to guarantee access to the information over a long period of time. On the other, human input is necessary in the form of archive description, work processes, and the use of standards to keep the information accessible and interpretable as long as possible. This is a task subjected to many risks, which a digital long-term preservation platform has to cover. Such risks, inherent to digital long-term preservation, can be grouped into five categories spanning a long period of time, as shown in Figure 13. This figure should not be interpreted as a ‘real' order of events or any causality between events. It just gives an indication of the risks an institution has to deal with over time, when preserving digital objects for the long-term. In this section we give a description of such risks, because they will define the technical requirements of our archive.

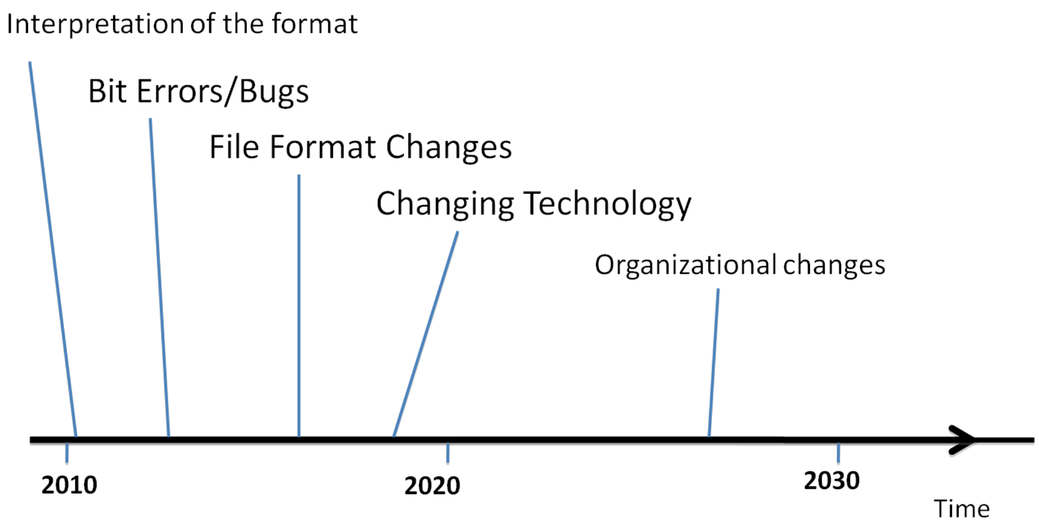


Figure 13: Risks of long-term preservation

It is the responsibility of the archive not only to keep the archived information intact, but also interpretable over time. Today, there is a big discrepancy between the short lifespan of file formats and the need for long-term preservation. File formats and their different flavours (e.g., TIFF, GeoTIFF, pyramid TIFF) emerge rapidly. One risk that the archive might first have to deal with is the right interpretation of the file format. Further ahead, file formats might become obsolete. The archive would then have two solutions for the presentation of stored information to the end-user: migration or emulation. Metadata is needed to support both actions.

The archive also has to cope with bit errors, bit rot and bugs. Bit rot is the natural decay of digital information and storage media over time, resulting in eventual unreadability. Bit rot affects different storage formats at different rates depending on the format's durability. Magnetic storage and optical discs are especially prone to varying forms of digital decay. For the time being, masked ROM cartridges appear to be fairly durable while EPROMs are at greater risk. This is why the risk of bit rot comes into play before the risk of file format obsolescence. The archive will need to have processes, and therefore also the metadata, in place to correct these errors and to guarantee the authenticity of the data. Examples of these are binary metadata, e.g., file format information, fixity information like MD5 checksums, and digital signatures. These metadata fields can help detect the loss of information, but only mirroring backups can prevent or cure it.

Yet further ahead, another threat for digital preservation platforms will be technological changes, as can be seen in Figure 13. Technological changes are less frequent than file format obsolescence. Examples of these are for instance Commodore 64 games, operating system incompatibilities, or even changing technologies regarding information storage, such as relational databases, graph databases, etc. This puts specific demands on the architecture of the archive: it has to organise its data in a platform independent manner. The OAIS reference model, discussed in the next section, provides high-level architecture to deal with this issue.

In the long run institution structures, terminologies, and the intended audience for one's information might change. In practice, this means that one's descriptive metadata can change, and the metadata format used for it. Another issue on the same level is that of the rights on an archived object or institution, which will change over time too. To keep the information interpretable, the archive needs:

- descriptive metadata, for a general description of the object, e.g., MARC
- rights metadata, for describing copyright statements, licenses, and possible grants that are given
- context metadata, for describing the relations of the content information to that of external data sources.

7.2 PREMIS

PREMIS (31) is a preservation standard based on the OAIS reference model (32), which is in fact provenance metadata supplemented with technical metadata and rights metadata to support preservation actions. This standard is currently in version 2.1, as the PREMIS Data Dictionary for Preservation Metadata. An XML schema is provided that implements the data dictionary for digital preservation. This preservation standard is described by a data model, which consists of five semantic units or classes important for digital preservation purposes:

- Intellectual Entities: part of the content that can be considered as an intellectual unit for the management and description of the content. This can be for example a book, a photograph, or a database.
- Object: a discrete unit of information in digital form, typically a multimedia object related to the intellectual entity.
- Event: An action that has an impact on an object or an agent.
- Agent: a person, institution, or software application that is related to an event of an object or is associated to the rights of an object.
- Rights: description of one or more rights, permissions of an object or an agent.

A new version is under development. It will change the data model to make intellectual entities another level of object, rather than remaining separate. Events and rights are directly related to an object, whereas an agent can only be related to an object through an event or through rights, as can be seen on Figure 14. In this way not only the changes to an object are stored, but also the event involved in such a change is described. These relationships offer the necessary tools to properly store the provenance of an archived object. The rights metadata needed for preservation are covered by the rights entity. Binary metadata, technical metadata, fixity metadata and structural metadata are encapsulated in the PREMIS data dictionary via the description of the object entity.

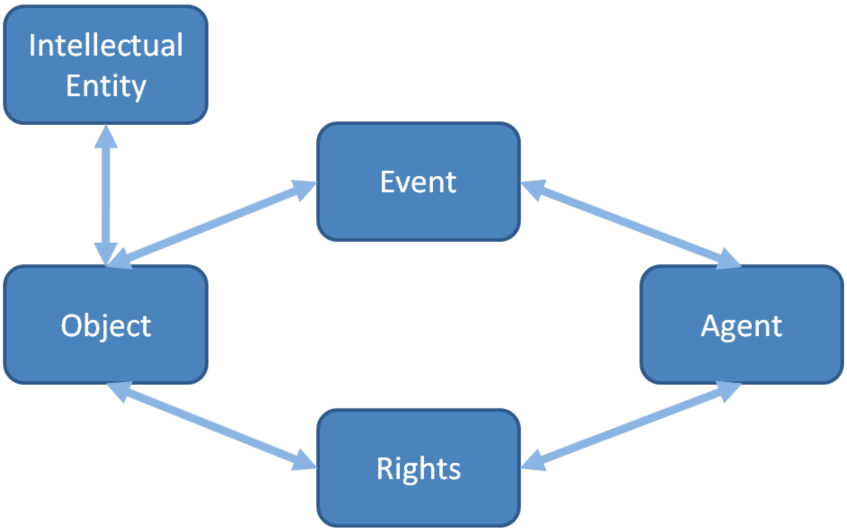


Figure 14: Overview of PREMIS

Bibliography

1. Dublin Core Metadata Initiative Homepage. [Online] http://dublincore.org/.
2. *SEPIA Data Element Set (SEPIADES).* [Online] http://marinemetadata.org/references/sepiades.
3. *Metadata Encoding and Transmission Standard.* [Online] 2009 йил 29-01. http://www.loc.gov/standards/mets/.
4. *HTML Specification.* [Online] http://www.w3.org/TR/html401/.
5. *Extensible Markup Language (XML) 1.0 (Fifth Edition).* [Online] http://www.w3.org/TR/REC-xml/.
6. *RDF Primer.* [Online] http://www.w3.org/TR/rdf-primer/.
7. *RDF/XML Syntax Specification (Revised).* [Online] http://www.w3.org/TR/rdf-syntax-grammar/.
8. *Turtle - Terse RDF Triple Language.* [Online] http://www.w3.org/TeamSubmission/turtle/.
9. *RDF Vocabulary Description Language 1.0: RDF Schema.* [Online] http://www.w3.org/TR/rdf-schema/.
10. *OWL Web Ontology Language .* [Online] http://www.w3.org/TR/owl-ref/.
11. *Anglo-American Cataloguing Rules.* [Online] http://www.aacr2.org/.
12. *Resource Description and Access.* [Online] http://www.rda-jsc.org/rda.html.
13. International standard bibliographic description (ISBD). [Online] http://archive.ifla.org/VII/s13/pubs/ISBD_consolidated_2007.pdf.
14. *Definition of the CIDOC Conceptual Reference Model.* [Online] http://www.cidoc-crm.org/docs/cidoc_crm_version_5.0.3.pdf.
15. *FUNCTIONAL REQUIREMENTS FOR BIBLIOGRAPHIC RECORDS.* [Online] http://archive.ifla.org/VII/s13/frbr/frbr_current_toc.htm.
16. *General International Standard Archival.* [Online] http://www.icacds.org.uk/eng/ISAD(G).pdf.
17. *MARC 21 Specifications for Record Structure, Character Sets, and Exchange Media.* [Online] http://www.loc.gov/marc/specifications/.
18. *LIDO - Lightweight Information Describing Objects.* [Online] http://www.lido-schema.org/schema/v1.0/lido-v1.0-specification.pdf.
19. *Categories for the Descriptions of Works of Art.* [Online] http://www.getty.edu/research/publications/electronic_publications/cdwa/index.html.
20. *SPECTRUM, the standard for collections management.* [Online] http://www.collectionstrust.org.uk/spectrum.
21. *Encoded Archival Description.* [Online] http://www.loc.gov/ead/.
22. *Art & Architecture Thesaurus.* [Online] http://www.getty.edu/research/tools/vocabularies/aat/about.html.
23. *Getty Thesaurus of Geographic Names.* [Online] http://www.getty.edu/research/tools/vocabularies/tgn/about.html.
24. *Union List of Artist Names.* [Online] http://www.getty.edu/research/tools/vocabularies/ulan/about.html.

25. *Library of Congress Subject Headings.* [Online] http://id.loc.gov/authorities/subjects.html.

26. *RKDartists&.* [Online] http://website.rkd.nl/Databases/RKDartists.

27. *Film identification - Enhancing interoperability of metadata - Element sets and structures.* [Online] http://filmstandards.org/fsc/index.php/EN_15907.

28. *Open Archives Initiative: Object Reuse and Exchange.* [Online] http://www.openarchives.org/ore/1.0/primer.html.

29. *Europeana Semantic Elements Specification.* [Online] http://version1.europeana.eu/c/document_library/get_file?uuid=77376831-67cf-4cff-a7a2-7718388eec1d&groupId=10128.

30. *Europeana Data Model.* [Online] http://group.europeana.eu/c/document_library/get_file?uuid=718a3828-6468-4e94-a9e7-7945c55eec65&groupId=10605.

31. *PREMIS Data Dictionary for Preservation Metadata.* [Online] http://www.loc.gov/standards/premis/v2/premis-dd-2-1.pdf.

32. *Reference Model for an Open Archival Information System.* [Online] http://public.ccsds.org/publications/archive/650x0b1.pdf.

33. *XML Schema.* [Online] http://www.w3.org/XML/Schema.