Harmonization methodology for metadata models

Mikael Nilsson, Ambjörn Naeve, Erik Duval, Pete Johnston, David Massart

Network of Excellence in Professional Learning

PROLEARN

European Commission Sixth Framework Project (IST-507310)

Deliverable D4.7 ***Harmonization of Metadata Standards***

Editor	*Mikael Nilsson*
Work Package	*Learning Objects, Metadata and Standards*
Status	*Final*
Date	*2008-01-21*

The PROLEARN Consortium

1. Universität Hannover, Learning Lab Lower Saxony (L3S), Germany
2. Deutsches Forschungszentrum für Künstliche Intelligenz GmbH (DFKI), Germany
3. Open University (OU), UK
4. Katholieke Universiteit Leuven (K.U.Leuven) / ARIADNE Foundation, Belgium
5. Fraunhofer-Gesellschaft zur Förderung der angewandten Forschung e.V. (FHG), Germany
6. Wirtschaftsuniversität Wien (WUW), Austria
7. Universität für Bodenkultur, Zentrum für Soziale Innovation (CSI), Austria
8. École Polytechnique Fédérale de Lausanne (EPFL), Switzerland
9. Eigenössische Technische Hochschule Zürich (ETHZ), Switzerland
10. Politecnico di Milano (POLIMI), Italy
11. Jožef Stefan Institute (JSI), Slovenia
12. Universidad Polictécnica de Madrid (UPM), Spain
13. Kungl. Tekniska Högskolan (KTH), Sweden
14. National Centre for Scientific Research “Demokritos” (NCSR), Greece
15. Institut National des Télécommunications (INT), France
16. Hautes Etudes Commerciales (HEC), France
17. Technische Universiteit Eindhoven (TU/e), Netherlands
18. Rheinisch-Westfälische Technische Hochschule Aachen (RWTH), Germany
19. Helsinki University of Technology (HUT), Finland
20. imc information multimedia communication AG (IMC), Germany
21. Open Universiteit Nederland (OU NL), Netherlands

Document Control

Title: **Harmonization of metadata standards**

Author/Editor: **Mikael Nilsson**

E-mail: **mikael@nilsson.name**

Amendment History

Version	Date	Author/Editor	Description/Comments
0.1	2007-06-10	Mikael Nilsson	Initial draft
0.2	2007-06-25	Mikael Nilsson	First round of feedback integrated
0.3	2007-07-28	Mikael Nilsson	Draft deliverable finalized
0.9	2008-12-13	Mikael Nilsson	Final draft to reviewers
1.0	2008-01-21	Mikael Nilsson	Final version

Harmonization of Metadata Standards

ProLEARN Deliverable D4.7
Final version, January 2008

Editor:
Mikael Nilsson <mikael@nilsson.name>

Contributors:
Ambjörn Naeve
Erik Duval
Pete Johnston
David Massart

Contents

1 INTRODUCTION 4

2 THE NOTION OF METADATA 5

3 METADATA STANDARDS 6

3.1 Domain-specific complete element sets and schemas 6

4 HARMONIZATION 6

4.1 Abstract Model standards 7

4.2 Vocabulary standards 8

4.2.1 Element vocabularies 9

4.2.2 Value vocabularies 10

4.3 Syntax standards 11

4.4 Application profiles 12

5 CHALLENGES FOR HARMONIZATION 13

5.1 Application Profiles in DC and LOM 13

5.2 Identifying and reusing elements 14

5.3 Requirements for application profiles 15

5.4 Summary of obstacles 15

6 ADDRESSING THE HARMONIZATION ISSUES 16

7 CONCLUSIONS 18

8 REFERENCES 19

1 Introduction

Metadata allows systems, applications and users to manage and access resources without a need for interaction with the resource itself. For this reason, the administration and exchange of metadata is a central activity in systems that manage learning objects. Metadata considerations are fundamental when creating interoperable e-learning tools, and metadata standards have been among the very first learning technology standards to mature.

However, despite enormous progress in the harmonization of learning object metadata standards, culminating in the release of the IEEE Learning Object Metadata standard in 2002, there remains a core of unsolved issues with respect to metadata interoperability and metadata harmonization. Today there is a plethora of metadata specifications (such as IEEE LOM, Dublin Core, METS, MODS, MPEG-7, etc), many of which are useful in whole or part for activities related to teaching and learning. While each specification in itself is designed to increase system interoperability, we are increasingly seeing systems that need to work with more than one of these specifications. Adding support for an additional specification generally presents a significant amount of added complexity in implementation. The reason for this is a lack of harmonization between specifications. In an ideal world, adding support for an additional metadata specification would be a simple matter of slightly extending the existing system.

Existing solutions to the metadata harmonization issue are few - systems are either limited to a single specification, or implement ad-hoc solutions that only work in that particular environment. There are many examples of "mappings" between specifications that provide partial solutions to the problem, but generally fail due to low-fidelity translations and lack of generality (i.e. the mapping only works for limited parts of specifications). Another solution is to create a top-level data model that encompasses the common aspects of all the specifications. This has proven to be feasible in relatively well-constrained domains such as resource aggregation, where the work on the RAMLET top-level ontology for resource aggregation has proceeded well within the IEEE. In the field of general metadata, where there is no such common ground, sucn an approach is substantially less likely to be successful.

This deliverable analyses a number of existing metadata specifications in order to isolate the reasons and issues behind harmonization problems. By making these issues explicit, we hope to contribute towards producing a benchmark against which possible solutions can be measured. The deliverable also discusses the potential of a solution based on harmonization of the various abstract models used in the metadata specifications.

The deliverable begins with a short introduction to metadata in Section 3. Section 4 discusses a set of metadata specifications that are highly relevant to learning and

teaching. Section 5 forms the core of the deliverable and analyzes the harmonization issues among a chosen set of specifications. Section 6 generalizes the analysis in Section 5 and makes a deeper analysis of the relationship between IEEE LOM and Dublin Core. Section 7, finally, points to possible ways to address the identified harmonization issues.

2 The notion of metadata

In practice, most modern metadata standards adopt a definition of metadata that allows descriptions of digital and non-digital things alike, usually collectively termed *resources*, but limits the type of metadata descriptions to a very restricted kind of metadata, as defined by the metadata standard.

In this deliverable, we will use the term “metadata” in the sense of the following modern definition:

Machine-processable information about (digital and non-digital) resources

This definition encompasses not only human-assigned information about a resource (such as name/title, subject and creator), but may also be used for information relating to e.g.:

- the life-cycle of a resource (different versions, history, etc.)
- technical aspects of a resource (size, format, functionality, etc.)
- relations between resources and aggregations of resources (lessons comprised of learning objects etc.)

It encompasses information not only about digital resources, but also about e.g.:

- learners and teachers (history, competencies, etc.)
- events (location, participants etc.)
- abstract notions (pedagogical designs, terms in taxonomies etc.)

In recent years, the notion of metadata has started to expand, taking new forms and being managed in new ways. For example, the following aspects go beyond the traditional notion of metadata as we have encoded in the metadata specifications of today:

- Automated generation of metadata, where metadata is generated as part of the creative process, or part of usage or context of resources, or inferred from the contents of a resource.
- Attention metadata, where information about users’ actions and progress in systems is captured automatically and processed for different purposes. Such metadata is not used for traditional descriptive purposes, but rather for expressing contextual relationships and supporting adaptive system behaviour.
- Truly subjective metadata, such as metadata about emotions, mood, opinions, arguments, ratings etc., as sometimes seen in social networks and instant messaging contexts.
- Collaborative tagging as a way of capturing user-generated classifications.

Many of these developments have not yet been formalized in metadata specifications, and as such will not form part of the analysis in this deliverable. However, any attempt

at harmonization of metadata standards must address these aspects of metadata as well in order to be prepared for future developments within this field.

3 Metadata standards

The terms "metadata standard" or "metadata schema" are often used to refer to the various kinds of specifications for metadata available from different organizations. Note that in the notion of "standard" we include both international de jure standards, as well as metadata specifications from established specification organizations.

Metadata standards come in different forms and with different kinds of audiences, and for the purpose of this deliverable it is useful to look at the following broad categories of standards.

Generic, framework-level models

- The DCMI Abstract Model (DCAM), defining the underlying model for Dublin Core metadata terms (Powell, Nilsson, Naeve, and Johnston 2004).
- RDF, the Resource Description Framework, a general-purpose, web-oriented metadata framework, defined by the W3C.

Generic, framework-level syntaxes

- Expressions of Dublin Core in RDF/XML/XHTML, describing syntaxes for encoding DCAM-compatible metadata in various syntaxes.
- RDF/XML and other RDF syntaxes.

General-purpose element sets to be reused in many different contexts

- DCMI Metadata Terms, defining a set of metadata terms conforming to the DCMI Abstract Model.

3.1 Domain-specific complete element sets and schemas

- IEEE LOM Data Model, defining the basic metadata elements and how they combine into a LOM instance. IEEE LOM currently has an XML syntax only.
- MODS, Metadata Object Description Schema - an XML schema for encoding MARC21 library records defined by the Library of Congress.
- MPEG-7 MDS, defining a complex XML format for multimedia metadata.

It should be obvious from this list that comparing "metadata standards" is not an easy task. In this deliverable, we will tackle the problem by analysing five "groups" of specifications:

1. The IEEE LOM family of specifications
2. The DCMI family of specifications
3. The RDF family of specifications
4. The MODS schema
5. The MPEG-7 specification

4 Harmonization

Learning object metadata interoperability refers to the ability of different systems to exchange information about resources. Metadata created in one system and then transferred to a second system will be processed by that second system in ways which are consistent with the intentions of the metadata creators (human or software).

Duval, Hodgins, Sutton, and Weibel (2002) set forth four fundamental principles for such interoperability, repeated in the Dublin Core – IEEE LTSC Memorandum of Understanding ("Memorandum", 2000). These are:

- **Extensibility**, or the ability to create structural additions to a metadata standard for application-specific or community-specific needs. Given the diversity of resources and information, extensibility is a critical feature of metadata standards and formats.
- **Modularity**, or the ability to combine metadata fragments adhering to different standards. Modularity is stronger than simple extensibility in that it requires that metadata from different standards, including metadata extensions from different sources, should be usable in combination without causing ambiguities or incompatibilities.
- **Refinements**, or the ability to create semantic extensions, i.e., more fine-grained descriptions that are compatible with more coarse-grained metadata, and to translate a fine-grained description into a more coarse-grained description.
- **Multilingualism**, or the ability to express, process and display metadata in a number of different linguistic and cultural circumstances. One important aspect of this is the ability to distinguish between what needs to be human-readable and what needs to be machine-processable.

Metadata *harmonization* as used in this deliverable refers to a further step beyond this level of system interoperability, and instead refers to interoperability *between metadata standards*. Harmonization then refers to the ability to use several different metadata standards in combination in a single software system. The rest of the deliverable will analyze the different groups of standards and try to find obstacles to harmonizations. In that analysis, the four interoperability principles above form a useful basis for evaluating the achieved progress in metadata harmonization.

In (Nilsson et al., 2007), a fifth principle is suggested, namely

- **Machine-processability**, or the ability to automate processing of different aspects of the metadata specifications, so that machines can handle extensions, manage modules, understand refinements and provide support for multilingualism.

This principle suggests that given the right support, harmonization may be realized in an automated fashion, with no need for translations, mappings or other manual interventions.

4.1 Abstract Model standards

Underlying most metadata specifications there is an assumption about an abstract model (sometimes referred to as "metamodel" or "data model"), within the framework of which the metadata is defined. The abstract model specifies the concepts used in the

standard, the nature of terms and how they combine to form a metadata description. The abstract model is the schematics used by an application to understand a metadata expression given in a specific format, thus making it possible for a single standard, though expressed in several different formats, to still be understood in a uniform way by users and applications.

Metadata elements are defined and presented using this model. In other words - the abstract model is the framework that exists independently of the particular metadata elements used. Abstract models are sometimes also the basis for query languages - just like SQL is dependent on the underlying relational database model.

In the above examples, we see several abstract models defined. IEEE LOM uses an abstract hierarchical model with no formal semantics. RDF, and as consequence, Dublin Core, use an entity-relationship model grounded in model-theoretical semantics, while MPEG-7 and MODS use an XML-based structure, to which MPEG-7 adds object-oriented semantics. The models differ substantially in their methods for adding extensions - the XML-based models base their extensions on XML Schema, IEEE LOM depends on being able to extend the hierarchy, while the entity-relationship-based models have no notion of "extensions" as there is no base set of elements to begin with.

Specification	Structure	Formal semantics	Extensions
IEEE LOM	Tree-based	No formal semantics	Additions to the tree
The DCMI specifications	Entity-relationship model	Model-theoretic semantics	Any entity or relation can be used
RDF	Entity-relationship model	Model-theoretic semantics	Any entity or relation can be used
MODS	XML tree	No formal semantics	XML Schema extensions
MPEG-7	XML tree	object-oriented	XML Schema and DDL (Data Definition Language) extensions

As is apparent from the many proposals about mapping from one model to another (MPEG-7 to DC, MPEG-7 to RDF/OWL, IEEE LOM to DC, MODS to DC, DC in MPEG-7), this plethora of models is making metadata harmonization a very difficult task.

4.2 Vocabulary standards

In order to fill the abstract models with concrete metadata elements, metadata vocabularies are needed. Nilsson et al. (2007) identifies two kinds of metadata vocabularies:

- **Element vocabulary** - a set of metadata elements, that are used as some form of "descriptive attribute" in a metadata record. Examples of elements include dcterms:creator (the Creator element from Dublin Core) and "General.Title" (the

title element from IEEE LOM). The corresponding element vocabularies would be the set of DCMI Metadata Terms and the set of IEEE LOM Data elements.

- **Value vocabulary** - a set of concepts or terms that can be used as value for a metadata element. Examples of such value vocabularies include the Library of Congress Headings, which includes terms such as "Biology" etc. Another example of a value vocabulary is the IEEE LOM Contributor Role vocabulary, containing terms such as "author", "illustrator" etc.

4.2.1 Element vocabularies

Element vocabularies and value vocabularies have fundamentally different characteristics. While value vocabularies are used to construct taxonomies and thesauri that describe relationships between concepts in terms of broader/narrower, containment etc, element vocabularies are used to construct schemas and ontologies that describe how metadata instances are to be constructed.

Abstract models tend to contain a model for describing element vocabularies. The following table summarizes a few important differences between the ways element vocabularies are handled in the different models. In particular, we highlight the method of defining element vocabularies, and the method for identifying elements in metadata instances.

The table also summarizes the ways that elements may reference other elements. Semantic models generally support refinement, i.e. defining elements that are more precise than an existing element (such as dcterms:creator being more precise than dcterms:contributor). Hierarchical models generally support structural relationships between elements.

Specification	Method for defining element vocabularies	Element identification	Element relationships
IEEE LOM	Defines element vocabularies by describing the placement of the elements at a particular node in the hierarchy describing the metadata instances.	Tree path	Does not allow for refinements of elements, but does allow sub-structures.
The DCMI specifications	Define element vocabularies using RDF Schema.	URI	Allow refinement using RDF Schema constructs.
RDF	Defines element vocabularies using RDF Schema.	URI	Allows refinement using RDF Schema constructs.
MODS	Defined as XML elements only	XML name	Does not allow for refinements of elements, but does allow sub-structures.
MPEG-7	Elements defined in MPEG-7 DDL (Data Definition Language).	XML name	Allows refinement through subclassing in DDL, as well as sub-

			structures.

It seems clear that element vocabularies are very problematic from a harmonization point of view, as elements in the different standards are defined, identified and related using fundamentally different underlying mechanisms. For example, an XML element and an RDF property have fundamentally different characteristics.

4.2.2 Value vocabularies

Value vocabularies are usually simply referred to in different ways in metadata instances. The following table summarizes how value vocabularies are defined and referenced in the different specifications.

Specification	Defining value vocabularies	Referring to values
IEEE LOM	IEEE LOM does not define a method for describing value vocabularies.	Refers to values using two string tokens: the "Source" and the "Value".
The DCMI specifications	Do not define a preferred method for defining value vocabularies, although SKOS is becoming more and more popular (see below).	Refer to values using URIs or natural language strings.
RDF	Does not define a preferred method for defining value vocabularies other than RDF Schema, although SKOS is becoming more and more popular (see below).	Refers to values primarily using URIs.
MODS	Has no way of defining vocabularies except listing them in the XML Schema.	Refers to values using natural language strings.
MPEG-7	Defines vocabularies by listing them in DDL.	Refers to values using natural language strings, unless they are XML elements, in which case there is a built-in reference mechanism.

SKOS (Simple Knowledge Organisation Systems) is a W3C working draft specification for defining taxonomies and classification schemes (see http://www.w3.org/2004/02/skos/).

The major harmonization issue with value vocabularies has to do with the way terms in the vocabulary are referenced in metadata instances. In the above table, there are four major methods used: URIs, Souce/Value pairs, string tokens and natural language strings. Different methods of identification imply different levels of precision, support for multilingualism and application independence. In order of decreasing precision (the examples are made up for illustration purposes only):

Value referencing method	Example	Ambiguity	Multilingualism	Application independence
URI	http://www.loc.gov/subjects/Biology	Depends on URI scheme used and identifier stability	fully multilingual	reusable across any kind of application
Source/value pair	Source: LCSH, Value: Biology	Depends on what "Source" token is used, as well as pre-agreement on allowed "source" token.	fully multilingual	reusable across any application
Token	EA32	Unique as long as it is tied to a particular XML schema or other context	fully multilingual	depends on knowledge of XML Schema/context
natural language string	Biology	Ambiguous	Not multilingual	Cannot be reused, as meaning is context-dependent

Clearly, URIs and source/value pairs are potent ways of referencing value vocabularies. See also CORES (Baker & Dekkers, 2002), an agreement to use URIs to identify components of metadata standards.

4.3 Syntax standards

Exchanging metadata records requires a serialization format, or *metadata syntax*.

Specification	Syntax
IEEE LOM	Can be expressed in XML, other syntaxes can be defined.
The DCMI specifications	Dublin Core metadata can be expressed using XML, any RDF syntax (see below) or HTML meta tags.
RDF	RDF can be expressed in RDF/XML, N3, Turtle, RDFa (**RDF** in HTML **a**ttributes) and a few other, specialized syntaxes.
MODS	As the model is based on XML, MODS can only be expressed in XML.
MPEG-7	Like MODS, only XML expression is possible.

Metadata syntaxes are strongly linked to the abstract models of the respective specifications. As can be seen, the specifications whose abstract models are tightly linked to XML are also restricted to be expressed in XML. In the case of IEEE LOM, the abstract model is not based on XML, but is still based on a similar hierarchy. This is one reason for the existence of only one syntax of LOM so far (Nilsson et al, 2003). While that does not make other syntaxes for these metadata specifications impossible, clearly the design of an abstract model can influence the complexity of expression using different syntaxes.

Why would it be important to be able to express information using many syntaxes? There are several, related reasons for this:

- Different syntaxes have different features. For example, XML provides an easily parsed, well-structured syntax in the cases that the data is relatively homogeneous, while RDF provides a more flexible model in the face of heterogeneous data. Thus different applications will support certain syntaxes better.
- Different syntaxes will support different query formalisms, which are useful in different contexts.
- Metadata might be stored not only in files using text-based syntaxes, but also in databases - or be accessed using programming interfaces. A standard that can more easily be bound to different formalisms will be easier to implement in various systems.

4.4 Application profiles

In order to support community-specific and regional needs, metadata standards generally support a notion of customization through application profiles. While the exact methods used vary from specification to specification, the customization generally encompasses selecting a set of metadata elements from one or several element vocabularies, possibly extending the base element vocabulary as defined in the specification using locally defined elements, and choosing a set of useful value vocabularies for use with these elements.

Enabling such customizations of metadata standards is one of the ultimate goals of metadata harmonization as we have described it in this chapter, since many such use-cases depend on being able to reuse metadata elements from different specifications. Reusing an element in this case means referencing a metadata element in a way that can be automatically understood by an application, without reference to the definition of the application profile itself. If application profile-specific handling is needed, the process is better referred to as mapping.

Application profiles rely on the interoperability features of the respective metadata standards. The metadata standards we have discussed use slightly different notions of application profiles. Combined with the differences in abstract models we have discussed previously, this produces significant barriers for the harmonization that application profiles have been designed to enable.

The following table summarizes how application profiles are defined in the different specifications. In some cases, an application profile can be expressed in a machine-processable format. The table also summarizes the support for reuse of metadata elements across application profiles.

Specification	Application Profile support	Machine-readable expression of Application Profiles	Reusability
IEEE LOM	Profiles defined as restrictions/extensions of the base schema.	Currently only possible through XML Schema.	Difficult to reuse extensions reliably as element vocabularies are not well-defined.
The DCMI specifications	Profiles defined as arbitrary restrictions of arbitrary combinations of elements.	Several proposed formats ("Guidelines", 2005, Description Set Profiles).	Any part of an application profile can be reused separately.
RDF	No notion of application profiles, though OWL ontologies sometimes fill a similar function.	No formalism except OWL for ontologies.	Fully reusable.
MODS	Profiles are defined as XML extensions.	XML Schema.	Difficult to reuse extensions, though XML namespaces could help.
MPEG-7	Profiles are defined as XML extensions.	MPEG-7 DDL (Data Definition Language).	Difficult to reuse extensions, though XML namespaces could help.

5 Challenges for harmonization

Let us now focus on the two major metadata specifications in the e-learning domain: IEEE LOM and Dublin Core, and the possibilities for harmonization.

5.1 Application Profiles in DC and LOM

The Dublin Core and LOM interpretations of the concept of application profile are both rooted in the corresponding abstract models underpinning these standards. A Dublin Core application profile refers to properties, vocabulary encoding schemes and syntax encoding schemes; a LOM application profile refers to LOM data elements or extended data elements and their value spaces, using the range of datatypes specified by the LOM standard.

As has already been discussed these are fundamentally different types of constructs: an occurrence of a LOM data element is interpreted through the semantics of the LOM abstract model, and a reference to a DC property is interpreted through the semantics of the DCMI abstract model. Neither approach is sufficient to support the Lego-like assembly of a modular metadata description which draws on both the LOM and DC metadata standards. Secondly, the LOM standard provides not only a set of data elements, but also a default pattern for the use of those data elements, a “base” application profile to which other community- or application-specific LOM application profiles should also conform.

Closely related to this second point is that the LOM abstract model does not define a mechanism for uniquely identifying and referencing data elements within a global context. While the use of extended data elements is possible, the disambiguation of those elements is reliably possible only within a context where the use of names is controlled. The LOM abstract model does not lend itself to the reuse of data elements within a global context, or to the sharing of LOM metadata descriptions beyond a context in which names are controlled.

The DC and LOM application profile constructs are both useful in formalising the way in which the implementers of metadata standards customise and (to a greater or lesser degree) extend those standards. They also provide a basis for disclosing existing work and encouraging the reuse of components used within existing application profiles, again subject to some limitations. They highlight that a degree of mixing and matching is indeed possible – but only within the framework of the corresponding abstract model. For DC and LOM, the incompatibility of those abstract models means that the two incompatible application profile constructs are not sufficient to address the problem of how to use component parts of those two standards in combination.

5.2 Identifying and reusing elements

As shown in Nilsson et al. (2007), mixing different metadata standards using the XML format does not work the way we would want it to. Using RDF as a common format works well with standards that use an abstract model compatible with RDF, but is still problematic for LOM and other standards based on an elements-in-elements model. The main reason for this is that such models have no canonical interpretation as entity-relationship models, and thus need to be reinterpreted/reengineered in order for them to be usable in RDF.

The CORES Resolution (Baker and Dekkers, 2002), which has been signed by both the IEEE LTSC and the Dublin Core Metadata Initiative, encouraged the owners of metadata standards to assign URI references to their “elements”, the “units of meaning comparable and mappable to elements of other standards”, but it did not specify what “comparable and mappable” meant. As a consequence the owners of different standards assigned URI references to "elements" that were created within different abstract models and used metadata formats that rely on those incompatible abstract models for their meaning and interpretation. The assignment of a URI reference to an "element" means that it can be unambiguously cited, but it does not change the nature of the "element". For example, it is not meaningful to use a URI reference for a LOM element as, e.g., a property URI in a Dublin Core metadata description. Similar incompatibilities have been noted between, e.g., RDF and MPEG-7 (van Ossenbruggen, Nack and Hardman, 2004 and Nack, van Ossenbruggen and Hardman, 2005).

The conclusion we may draw from this analysis is that we must not confuse the components used in a metadata format with the constructs in the abstract model. The components in a metadata format, such as “element URIs” may seem to be similar and compatible, but in reality they belong to completely different frameworks that might not be compatible. There are several problematic scenarios, including:

- Mixing two metadata formats created to conform to different abstract models, such as Dublin Core XML and LOM XML. A similar example is trying to use parts

of a Dublin Core RDF description serialized in the RDF/XML language together with elements from another XML language such as the LOM XML language. As LOM and RDF use incompatible abstract models, this also leads to 'nonsensical' metadata constructs (Johnston, 2005).

- Reusing metadata terms or elements adhering to different abstract models, regardless of the metadata format used, such as reusing a Dublin Core element URI in a LOM metadata description. As discussed in Nilsson et al. (2007), this leads to nonsensical metadata constructs, since the URIs of Dublin Core and of LOM must be interpreted in terms of different abstract models. For example, the Dublin Core XML expression forces an interpretation of XML elements as properties - an interpretation that may not apply to included LOM metadata elements expressed in XML.
- Mixing two different syntaxes expressing the same specification, when those two expressions apply different interpretations to the use of similar components in the metadata format. This is the case with the Dublin Core XML binding, which must be interpreted using a different set of rules than the RDF/XML serialization of the Dublin Core RDF binding, although they contain component parts that are confusingly similar.

Hence we must conclude that the notion of "reusing elements" between metadata standards and formats using incompatible abstract models is fundamentally flawed. While assigning URI references for the component parts of a metadata standard is clearly a worthwhile effort in other ways, this does not really address the fundamental issue when creating interoperable metadata standards, namely the compatibility of their respective abstract models.

5.3 Requirements for application profiles

In conclusion, we see that in order to reuse components of different standards in a machine-processable way as discussed above, the following criteria must be met:

1. The components must be unambiguously identified, so that components from different sources can be clearly distinguished and their origins can be separated. This is addressed by the CORES resolution.
2. The components must adhere to compatible abstract models. There is currently no resolution to address this, although the Dublin Core – IEEE Memorandum of Understanding ("Memorandum", 2000) points in this direction.
3. A metadata format must be used that allows for consistent interpretation of the components with respect to their respective abstract models. This too is mentioned in the "Memorandum", but has yet to be realized.

5.4 Summary of obstacles

The following harmonization obstacles were raised in the previous section:

Extensibility

> Different abstract models have different methods for extensions, rendering the extensions mutually incompatible, and therefore not reusable across specifications.

Modularity

Different notions of application profiles lead to impossibility of combining fragments from different specifications.

Refinements

Not all abstract models support refinements, meaning that cross-vocabulary refinements becomes impossible.

Multilingualism

In the cases where abstract models do not clearly separate natural language items from abstract tokens, multilingualism quickly becomes an issue. This is handled well in at least LOM and Dublin Core.

Machine-processability of standards and extensions

This depends on articulated abstract models with well-defined semantics, which is currently provided by at least Dublin Core, RDF and MPEG-7.

Identification of elements and values

Clearly, a common model for referring to terms from element and value vocabularies is needed.

Syntaxes

The syntaxes used must be firmly rooted in the applicable abstract model.

Application Profiles

A common, formal model for Applications profiles is needed.

6 Addressing the harmonization issues

The above analysis shows that there are many difficulties on the road towards metadata harmonization. This chapter outlines a roadmap towards metadata harmonization generally, and between LOM and Dublin Core in particular. Five areas of harmonization are identified: identification harmonization, abstract model harmonization, vocabulary harmonization, application profile harmonization and syntax harmonization.

Issue	Comment	Needed actions
Identification	The first important issue to be resolved is that of identification, of both metadata elements and values taken from vocabularies. The analysis above shows that the tokens work locally and in well-defined communities, but on a global scale, global identifications is necessary. A related issue is when element identification depends on the placement of an element in a hierarchy, as in the LOM standard.	• Encourage the specification of URIs for values in controlled vocabularies. • Provide mappings from such URIs to tokens and natural language strings. • Encourage the specification of URIs for metadata elements.

Issue	Comment	Needed actions
Abstract Model	As has been shown above, value identification is relatively unproblematic, while element identification relies on understanding precisely *what* is being identified. In order for element identification to have an effect on harmonization, the elements need to be of the same kind, using a common understanding of the underlying model.	• Encourage harmonization through synchronization of abstract models. As we have seen in the analysis above, differences in abstract models create unnecessary incompatibilities. • Avoid relying on mapping of instance data for harmonization. As described in Nilsson et al (2007), except for highly similar standards, this creates a *m* x *n* problem, where every standard needs to mapped to every other. Instead try to align on the abstract model level. • Discourage the introduction of new abstract models into the domain, as this further fragments the community. Of particular worry is the work in ISO/IEC JTC1 SC36 on Metadata for Learning Resources. • In the cases where such synchronization is unfeasible, try to provide mappings on the level of abstract models.
Vocabulary model	While there is no strong requirement for a common value vocabulary model (since value identification is the major issue), a common model for element vocabularies is tightly linked to the harmonization of abstract models. Common, machine-understandable formats for element vocabularies are a prerequisite for enabling modularity - since this will enable automatic disassembling and processing of composite metadata.	In the analyzed specifications, three vocabulary models are used: RDF Schema, XML Schema and MPEG-7 DDL. Relying on a syntax-oriented model such as XML Schema to define abstract entities that can be reused across syntaxes and systems leads to difficult interoperability issues. Therefore the recommendation is to define element vocabularies using RDF Schema, even if RDF itself might not always be used as a way of expressing the metadata.

Issue	Comment	Needed actions
Application Profile Model	Working cross-standard application profiles require a common understanding of what an application profile is. This is dependent on the issues above, in particular regarding identifying and defining element vocabularies. If we are to support the multitude of description types mentioned in the beginning of this paper, an application profile model cannot be based on a "base" model such as LOM, as this would render the model unusable for describing other things than e.g. learning objects.	Models for application profiles that are independent of a particular element set need to be developed. As the LOM example shows, such models must be able to handle high structural complexity and specificity.
Syntaxes	A syntax is useless without a processing model, and such a model must be based on the abstract model of the metadata standard.	Make sure metadata syntaxes are firmly grounded in an abstract model, and that, conversely, the abstract model is considered before the syntax when developing metadata specifications.

7 Conclusions

In this deliverable we have analyzed the obstacles to metadata harmonization. The issues fall in three broad categories:

Conventions

> The different metadata specifications use different methods for identifying and describing metadata elements and terms from value vocabularies. It seems possible to enable high fidelity harmonization solutions to these issues without disrupting the existing specifications. For example, terms identified by source/value pairs can be assigned URIs.

Models

> The specifications differ substantially in how they define metadata records, and in how metadata is structured and processed. A mapping solution is therefore destined to be incomplete and suffer from not being generalizable to extensions. For example, the IEEE LOM notion of "Category" has no correspondence in Dublin Core metadata, and any generalizable mapping of Categories will therefore be problematic.

Combinations

> Combining elements to form application profiles, and encoding them in syntaxes are both processes that rely heavily on models as well as conventions. It is likely that once conventions and models are harmonized, applications profiles and syntaxes will become more easily addressable harmonization issues.

The above three categories also represent milestones on a roadmap to harmonization - harmonize conventions, then models, then application profiles and syntaxes. It should

therefore be clear that a solution to the harmonization problems needs to take the whole framework of conventions, models and application profiles and syntaxes into account.

Recently, there has been a clear movement towards conventions based on Web architecture, leading to a strong recommendation for basing identification on URIs. There is also increased momentum towards describing element and value vocabularies in a Web architecture-friendly way, using the RDF Vocabulary Description language (RDF Schema) for element vocabularies and SKOS (Simple Knowledge Organization Systems) for describing value vocabularies, i.e. controlled vocabularies, taxonomies and classification schemes.

For abstract models, a consensus has yet to be reached, although the Resource Description Framework (RDF) does provide a framework well founded in Web architecture and a formal semantics. This deliverable still recommends that metadata specifications harmonize their models with the RDF model and, by extension, the semantic web.

For application profiles and syntaxes, no firm guidances can really be given, though developments such as ontologies and the Dublin Core Description Set Profile specification remain highly relevant.

Concrete work on harmonizing IEEE LOM and Dublin Core is currently progressing within the Joint DCMI / IEEE LTSC Taskforce[1]. The approach taken is that of reinterpretation of the IEEE LOM data elements in terms of a completely different abstract model – the DCMI Abstract Model. The resulting specifications make LOM elements reusable in Dublin Core and RDF metadata, but at the cost of imperfect translation. At the same time, work on a “new LOM” is slowly starting, although it is unclear at this point what approach towards harmonization that will be taken.

In a similar spirit, the RDA (Resource Discovery and Access)[2] project is redesigning the data model that is behind the world's library data specifications (the MARC data model). The work is done in collaboration with the Dublin Core Metadata Initiative, and aims to produce a model that is compatible with the DCMI Abstract Model and RDF[3].

Together, these two initiatives, both of which include important contributions from ProLEARN members, demonstrate important progress towards harmonization of several important metadata domains – generic metadata using Dublin Core, educational metadata, and library metadata, as well as a widening from the all-digital domain to the domain of physical artefacts (books).

Harmonizing metadata specifications in the way outlined in this document seems an overwhelming task, but the steady flow of important developments still makes the future seem bright.

[1] http://dublincore.org/educationwiki/DCMIIEEELTSCTaskforce

[2] http://www.collectionscanada.gc.ca/jsc/rda.html

[3] http://dublincore.org/dcmirdataskgroup/

8 References

Baker, T. & Dekkers, M., (2002), CORES Standards Interoperability Forum Resolution on Metadata Element Identifiers. http://www.cores-eu.net/interoperability/cores-resolution/

Dublin Core Application Profile Guidelines (2003), CEN Workshop Agreement CWA 14855. ftp://ftp.cenorm.be/PUBLIC/CWAs/e-Europe/MMI-DC/cwa14855-00-2003-Nov.pdf

Duval, E., Hodgins, W., Sutton, S. & Weibel, S. L. (2002), Metadata Principles and Practicalities, D-Lib Magazine, April 2002. http://www.dlib.org/dlib/april02/weibel/04weibel.html

Friesen, N., Mason, J. & Ward, N. (2002), Building Educational Metadata Application Profiles, Dublin Core - 2002 Proceedings: Metadata for e-Communities: Supporting Diversity and Convergence. http://www.bncf.net/dc2002/program/ft/paper7.pdf

Godby, C. J., Smith, D. & Childress, E. (2003), Two Paths to Interoperable Metadata, Proceedings of DC-2003: Supporting Communities of Discourse and Practice – Metadata Research & Applications, Seattle, Washington (USA). http://www.siderean.com/dc2003/103_paper-22.pdf

Guidelines for machine-processable representation of Dublin Core Application Profiles (2005), CEN Workshop Agreement CWA 15248. ftp://ftp.cenorm.be/PUBLIC/CWAs/e-Europe/MMI-DC/cwa15248-00-2005-Apr.pdf

Heery, R. & Patel, M. (2000), Application Profiles: mixing and matching metadata schemas, Ariadne Issue 25, September 2000. http://www.ariadne.ac.uk/issue25/app-profiles/

Johnston, P., (2005a), XML, RDF, and DCAPs. from http://www.ukoln.ac.uk/metadata/dcmi/dc-elem-prop/

Memorandum of Understanding between the Dublin Core Metadata Initiative and the IEEE Learning Technology Standards Committee (2000). http://dublincore.org/documents/2000/12/06/dcmi-ieee-mou/

Nack, F., van Ossenbruggen, J. & Hardman, L. (2005), That obscure object of desire: multimedia metadata on the Web, part 2, IEEE Multimedia 12 (1) 54-63. http://ieeexplore.ieee.org/iel5/93/30053/01377102.pdf?arnumber=1377102

Nilsson, M., Palmér, M. & Brase, J. (2003), The LOM RDF Binding - Principles and Implementation, Proceedings of the Third Annual ARIADNE conference. http://kmr.nada.kth.se/papers/SemanticWeb/LOMRDFBinding-ARIADNE.pdf

Nilsson, M., Johnston, P., Naeve, A., Powell, A. (2007), The Future of Learning Object Metadata Interoperability, in Harman, K., Koohang A. (eds.) Learning Objects: Standards, Metadata, Repositories, and LCMS (pp 255-313), Informing Science press, ISBN 8392233751.

Powell, A., Nilsson, M., Naeve, A., Johnston, P. Dublin Core Metadata Initiative Abstract Model, DCMI recommendation, 2007, http://dublincore.org/documents/abstract-model/

van Ossenbruggen, J., Nack, F. & Hardman, L. (2004), That obscure object of desire: multimedia metadata on the Web, part 1, IEEE Multimedia 11 (4) 38-48. from http://ieeexplore.ieee.org/iel5/93/29587/01343828.pdf?arnumber=1343828