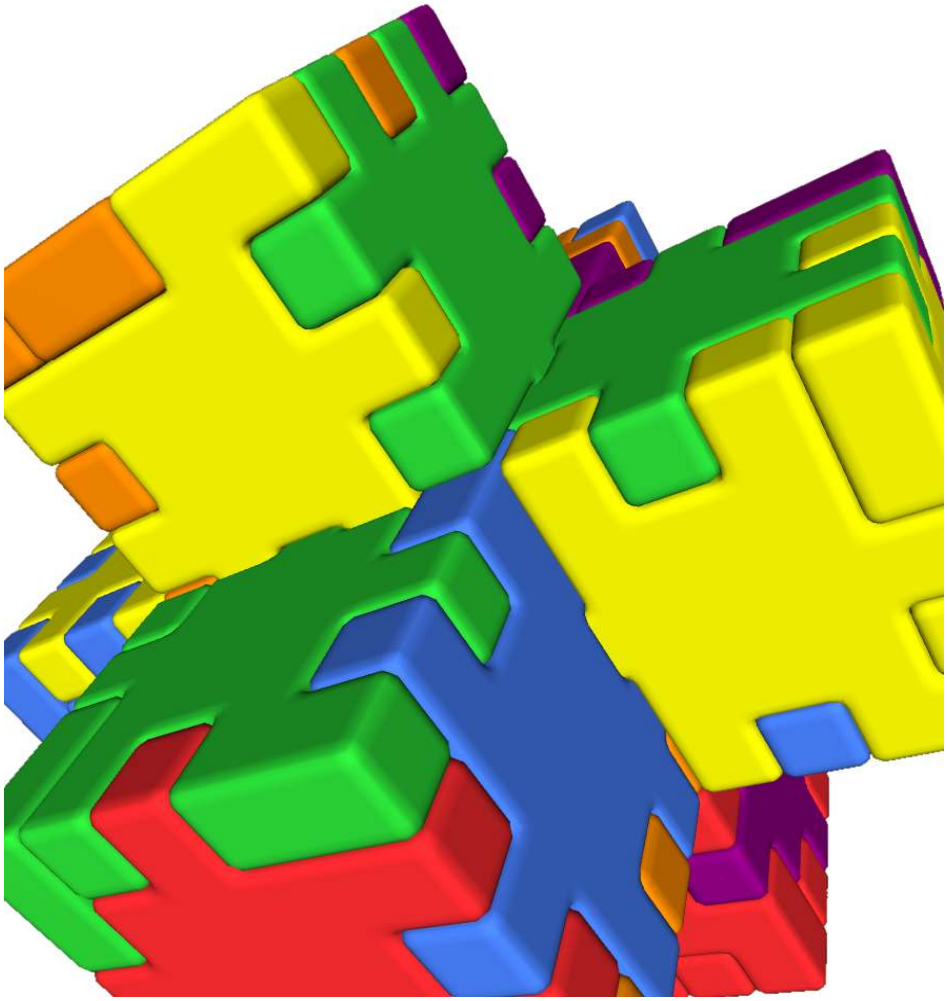
KTH Computer Science
and Communication

From Interoperability to Harmonization in Metadata Standardization

Designing an Evolvable Framework for Metadata Harmonization

MIKAEL NILSSON

Doctoral thesis
Stockholm, Sweden 2010



TRITA-CSC-A 2010:15
ISSN 1653-5723
ISRN KTH/CSC/A–10/15–SE
ISBN 978-91-7415-800-7

KTH School of Computer Science and Communication
SE-100 44 Stockholm
SWEDEN

Akademisk avhandling som med tillstånd av Kungliga Tekniska Högskolan framlägges till offentlig granskning för avläggande av teknologie doktorsexamen i medieteknik onsdagen den 15 december 2010 klockan 13.00 i Sal F3, Lindstedsvägen 26, KTH Campus, Stockholm.

Cover image created with Happy Solver: http://happysolver.sourceforge.net/

Abstract

Metadata is an increasingly central tool in the current web environment, enabling large-scale, distributed management of resources. Recent years has seen a growth in interaction between previously relatively isolated metadata communities, driven by a need for cross-domain collaboration and exchange. However, metadata standards have not been able to meet the needs of interoperability between independent standardization communities. For this reason the notion of metadata *harmonization*, defined as interoperability of combinations of metadata specifications, has risen as a core issue for the future of web-based metadata.

This thesis presents a solution-oriented analysis of current issues in metadata harmonization. A set of widely used metadata specifications in the domains of learning technology, libraries and the general web environment have been chosen as targets for the analysis, with a special focus on Dublin Core, IEEE LOM and RDF. Through active participation in several metadata standardization communities, a body of knowledge of harmonization issues has been developed.

The thesis presents an analytical framework of concepts and principles for understanding the issues arising when interfacing multiple standardization communities. The analytical framework focuses on a set of important patterns in metadata specifications and their respective contribution to harmonization issues:

- Metadata syntaxes as a tool for metadata exchange. Syntaxes are shown to be of secondary importance in harmonization.
- Metadata semantics as a cornerstone for interoperability. This thesis argues that the incongruences in the interpretation of metadata descriptions play a significant role in harmonization.
- Abstract models for metadata as a tool for designing metadata standards. It is shown how such models are pivotal in the understanding of harmonization problems.
- Vocabularies as carriers of meaning in metadata. The thesis shows how portable vocabularies can carry semantics from one standard to another, enabling harmonization.
- Application profiles as a method for combining metadata standards. While application profiles have been put forward as a powerful tool for interoperability, the thesis concludes that they have only a marginal role to play in harmonization.

The analytical framework is used to analyze and compare seven metadata specifications, and a concrete set of harmonization issues is presented. These issues are used as a basis for a metadata harmonization framework where a multitude of metadata specifications with different characteristics can coexist. The thesis concludes that the Resource Description Framework (RDF) is the only existing specification that has the right characteristics to serve as a practical basis for such a harmonization framework, and therefore must be taken into account when designing metadata specifications. Based on the harmonization framework, a best practice for metadata standardization development is developed, and a roadmap for harmonization improvements of the analyzed standards is presented.

Acknowledgements

Most importantly, I want to thank my loving partner Anna and my wonderful children for having not only endured the process of producing this thesis, but also provided much needed moral support.

I want to thank my supervisor Ambjörn Naeve, who has been an unending source of inspiration, support, ideas and projects ever since our first encounter more than a decade ago.

I also want to thank my colleague and friend Matthias Palmér, who has worked closely with me on many of the projects leading up to this thesis, and has been an irreplaceable discussion partner in the analysis of the theoretical and practical issues encountered.

The whole Knowledge Management Research Group at the Royal Institute of Technology, in particular Fredrik Paulsson, Hannes Ebner and Fredrik Enoksson have provided the fertile ground and engaging environment on which this work has grown. Thank you!

Mia Lindegren with team at the Uppsala Learning Lab has provided an important environment for developing the content of the thesis. Thank you!

I also want to thank a large number of international colleagues in the metadata community for enduring my theoretical excursions over the years, in particular Tom Baker, Andy Powell, Pete Johnston from DCMI, Erik Duval and Wayne Hodgins from IEEE LTSC and Erlend Øverby and Peter Karlberg from ISO/IEC JTC1 SC36. Many more have provided inputs of various kinds to the results in this thesis, and all have welcomed me as an active participant in their respective communities.

Many thanks also to Nils Enlund, Marko Turpeinen and Yngve Sundblad, professors at KTH, for helpfully supporting this research in more than one way over the years. Thanks also to Ingrid Melinder, dean at CSC, for being a great facilitator!

A special thanks to Stiftelsen för Internetinfrastruktur (.SE) and TeliaSonera, that have provided part of the funding for the research in this thesis.

Acronyms

DCAM *DCMI Abstract Model* – an abstract model for metadata used by the Dublin Core Metadata Initiative – http://dublincore.org/documents/abstract-model/

DCMI *Dublin Core Metadata Initiative* – a non-profit organization engaged in the development of interoperable metadata standards – http://dublincore.org/

DDL *Description Definition Language* – a part of the MPEG-7 standard that enables the definition of MPEG-7-compatible metadata schemas.

DSP *Description Set Profile* – a machine-processable expression of the metadata constraints of a Dublin Core Application Profile – http://dublincore.org/documents/dc-dsp/

FRBR *Functional Requirements for Bibliographic Records* – a conceptual model for metadata for library resources – http://www.ifla.org/en/publications/functional-requirements-for-bibliographic-records

GRRDL *Gleaning Resource Descriptions from Dialects of Languages* – a W3C specification for automatically extracting RDF triples from XML languages – http://www.w3.org/TR/grddl/

ILOX *Information for Learning Object eXchange* - an IMS Global Learning Consortium specification for describing learning object using a FRBR-compatible adaptation of IEEE LOM – http://www.imsglobal.org/LODE/

IMS *IMS Global Learning Consortium* – an organization producing learning technology specifications – http://www.imsglobal.org/

KIF *Knowledge Interchange Format* – a knowledge representation language – http://logic.stanford.edu/kif/

LODE *Learning Object Discovery and Exchange* - an IMS Global Learning Consortium specification for the discovery and retrieval of learning objects stored across more than one collection – http://www.imsglobal.org/LODE/

LOM *Learning Object Metadata* – an IEEE standard for metadata descriptions of learning objects

MARC *MAchine-Readable Cataloging* – a Library of Congress standard for representation and communication of bibliographic and related information – http://www.loc.gov/marc/

METS *Metadata Encoding and Transmission Standard* – a Library of Congress standard for XML encoding of descriptive, administrative, and structural metadata for library systems – http://www.loc.gov/standards/mets/

MLR *Metadata for Learning Resources* – an ISO metadata standard in development.

MODS *Metadata Object Description Schema* – a Library of Congress standard for XML encoding of selected data from MARC records – http://www.loc.gov/standards/mods/

OAI *Open Archives Initiative* – a organization producing repository interoperability specifications – http://www.openarchives.org/

OWL *Web Ontology Language* – a modeling language for expressing formal semantics of RDF properties and classes.

RDA *Resource Description and Access* – a specification based on FRBR specifying a set of instructions for the cataloging of books and other library materials – http://www.rda-jsc.org/rda.html

RDF *Resource Description Framework* – a W3C specification for metadata descriptions – http://www.w3.org/RDF/

RIF *Rules Interchange Format* – an W3C specification for describing inference rules for RDF metadata

RSS *Really Simple Syndication* – a family of XML formats used to publish frequently updated content

SKOS *Simple Knowledge Organization System* – a W3C specification for represent knowledge organization systems such as thesauri or taxonomies using RDF – http://www.w3.org/2004/02/skos/

SPARQL *SPARQL Protocol and RDF Query Language* – a W3C query language for RDF – http://www.w3.org/TR/rdf-sparql-query/

UML *Unified Modeling Language* – a multipurpose, graphical, object-oriented modeling language – http://www.uml.org/

URI *Universal Resource Identifier* – a globally unique identifier designed to be used on the WWW.

VDEX *Vocabulary Description and EXchange Language* - an IMS Global Learning Consortium specification for exchanging definitions of value vocabularies for IEEE LOM and other metadata specifications – http://www.imsglobal.org/vdex/

XML *eXtensible Markup Language* – a W3C specification for encoding documents in machine-readable form – http://www.w3.org/XML/

XSL *eXtensible Stylesheet Language* – an XML-based language for transforming and rendering XML documents – http://www.w3.org/Style/XSL/

Contents

List of figures and examples

Included Papers

Paper 1

Nilsson, M., Palmér, M., Naeve, A. (2002), Semantic Web Meta-data for e-Learning - Some Architectural Guidelines, Proceedings of the 11th World Wide Web Conference (WWW2002), Hawaii, USA.

Paper 2

Nilsson, M., Palmér, M., Brase, J. (2003), The LOM RDF Binding - Principles and Implementation, Proceedings of the Third Annual ARIADNE conference.

Paper 3

Nilsson, M. (2004), The Edutella P2P Network - Supporting Democratic E-learning and Communities of Practice, in McGreal, R. (ed.) Online education using learning objects, Falmer Press, New York, 2004, ISBN 0-415-33512-4.

Paper 4

Nilsson, M., Johnston, P., Naeve, A., Powell, A. (2006), Towards an Interoperability Framework for Metadata Standards, Proceedings of the International Conference on Dublin Core and Metadata Applications, Manzanillo, Colima, Mexico 3 - 6 October 2006

Paper 5

Nilsson, M., Miles, A. J., Johnston, P., Enoksson, F. (2007), Formalizing Dublin Core Application Profiles - Description Set Profiles and Graph Constraints, in Sicilia M-A., Lytras, M. D. (Eds.): Metadata and Semantics, Post-proceedings of the 2nd International Conference on Metadata and Semantics Research, MTSR 2007, Corfu Island in Greece, 1-2 October 2007. Springer 2009

Paper 6

Nilsson, M., Naeve, A. (2010), Metadata harmonization: a roadmap for standardization, submitted for publication.

A summary of each paper and the papers themselves can be found at the end of this thesis.

1. Introduction

1.1 Metadata, Standards, Interoperability and Harmonization

The theme of this thesis is the complex nature of *metadata* in the context of a set of metadata *specifications* that have been developed, standardized and implemented specifically for use in the Internet environment.

Metadata can be informally defined as “data about data”, i.e., any kind of information that in some way references or describes aspects of some other piece of information. Metadata is introduced in information management systems in order to support certain administrative operations, including searching, displaying summaries or configuring interfaces. In essence, metadata creates a level of indirection, allowing systems to manage resources without ever having to delve into their physical or digital internals. Metadata can consist of all kinds of information about an item, ranging from its title, textual descriptions and subject classifications to accessibility characteristics and the contextual relationships between the described item and other things.

The core value proposition of metadata is that using metadata enables systems, applications and users to manage and access items without any need for direct interaction with the item itself (see Lytras & Sicilia, 2007). For this reason, the administration and exchange of metadata is a central activity in many systems that manage digital and non-digital objects, such as content management systems, learning object repositories and libraries.

Metadata specifications and standards add additional value by lowering the threshold for developing systems that exchange, reuse and combine metadata from different sources. A common standard ensures better documentation, more widespread know-how and better access to reusable tools. This is the core value proposition of metadata *interoperability*.

Realizing the potential inherent in the informed use of interoperable metadata requires large-scale coordination between the relevant actors in a field of practice. Metadata specifications tend to be designed for a particular community, with more or less well-defined items to be described and common usage scenarios.

This thesis will analyze modern metadata specifications from three main domains: educational technology, libraries, and generic web metadata, with a particular focus on IEEE LOM, Dublin Core and RDF.

In the field of educational technology, metadata considerations are fundamental when creating interoperable e-learning tools, and metadata standards have been among the very first learning technology standards to mature. For example, learning object metadata may be used by cataloging software for indexing, by learning management systems for matching learners with relevant resources, and by content players that configure the learning object to the user's environment and needs. But despite enormous progress in the harmonization of learning object metadata standards, partly though the work in the IMS Global Learning Consortium (IMS Global, 2004) , and building on the release of the IEEE Learning Object Metadata (IEEE Computer Society, 2002) standard in 2002, there remains a considerable amount of unsolved issues with respect to metadata interoperability. Some but not all of those issues are being addressed by the recent developments in ISO on the standard Metadata for Learning Resources (ISO/IEC 19788-1, 2009).

In the library domain, metadata in the form of cataloging has been an issue since the early days of public libraries. As library data is gradually being opened up to the rest of the world, major metadata interoperability issues are surfacing. The development of a new cataloging standard, Resource Description and Access (RDA) (Coyle and Hillmann, 2007), is right at the focal point of library metadata and interoperability, highlighting the complex situation with a multitude of metadata standards in use in the library world, such as the arcane MARC[1], and the XML[2]-based METS[3] and MODS[4] format.

Both libraries and educational technology touch the fields of web-oriented metadata, where the Resource Description Framework (RDF) (Klyne & Carroll, 2004) has been making important progress over the last decade, together with a growing stack of specifications supporting the Semantic Web, such as the Web Ontology Language (OWL) (World Wide Web Consortium, 2009). The Dublin Core Metadata Initiative[5] and its associated metadata specifications are often seen as the core of web metadata, but it has intricate relationships to both RDF and the library and educational domains. Multimedia metadata, as defined by MPEG-7 (ISO/IEC 15938-2:2002), is another high-profile metadata domain with its own set of conventions and principles.

When metadata designed according to different specifications from different domains meet, for example when communities evolve to increase their interaction, considerable difficulties in metadata management tend to arise (Chan & Zeng, 2006, Zeng & Chan, 2006). More often than not, their respective metadata specifications are, in one way or another, incompatible. The result is that the benefits of metadata interoperability within one standard are lost when standards are combined, development costs increase, systems fail to communicate and ad hoc, non-reusable solutions are introduced. Godby, Smith & Childress (2003) argue, based on experiments with metadata crosswalks, that “complete translations are possible only within a given community of practice, while only partial translations are possible between them”, They give the example of

1 MAchine-Readable Cataloging, a widely used library metadata standard maintained by the Library of Congress, with roots in the 1960s.

2 Extensible Markup Language, a W3C specification for defining markup languages.

3 Metadata Encoding and Transmission Standard, a metadata packaging format maintained by the Library of Congress.

4 Metadata Object Description Standard, an XML encoding of MARC, maintained by the Library of Congress

5 http://dublincore.org

library metadata, which we can assume to be fully combinable between different library metadata specifications, while a successful and complete combination with a learning technology metadata specification such as SCORM[6] is unlikely.

With the increasing ubiquity of Internet-based applications and cross-domain collaboration, such interoperability failures are destined to occur with increasing frequency. To counter the effects of interoperability failures, considerable efforts have been spent on *harmonization* of metadata standards[7], with the goal of increasing metadata interoperability across multiple metadata specifications.

1.2 The Purpose of this Thesis

This thesis describes the theoretical conclusions of several metadata harmonization initiatives, in the context of international metadata standardization activities in the fields of learning and teaching, libraries and web metadata. A number of major difficulties encountered when trying to use such metadata in combination is explored and a number of developments that might lead to solutions to the problems are presented.

At a first glance, the major problem of metadata interoperability seem to be about formats: the different standards all use different methods of encoding their information. Nowadays, many standards use XML-based encodings, but using XML is not a guarantee for interoperability. This thesis examines the complex issues arising from the use of different syntaxes, such as XML, RDF and HTML meta tags.

Even if the syntax issue could be addressed, many issues still remain. Some standards, such as Dublin Core, rely on an abstract framework that fits into many syntaxes. The purpose and usage of abstract models for metadata and how they support metadata interoperability is analyzed in this thesis.

Underlying formats and abstract frameworks is the subtle notion of semantics. With the rise of the RDF and the Semantic Web initiative of the W3C, the semantics of metadata descriptions has received increasing attention. This thesis tries to find an explanation for why semantics is a central aspect of metadata interoperability and harmonization, and to understand the implications for metadata standardization activities.

Setting formats and semantic issues aside, the thesis analyzes what it means to combine metadata from different standards. Many metadata implementations are derived from a core standard through the use of so-called application profiles. However, a closer dissection of the notion of application profiles reveals several incompatible definitions that are, in themselves, one cause of harmonization issues between standardization communities. This thesis tries to isolate the problematic factors in application profile harmonization.

The lessons learned from the analysis of formats, semantics and application profiles lead to implications for metadata standards. Many standards in use today are unnecessarily complex, unnecessarily incompatible and would benefit from a redesign based on best practices for harmonization. This thesis tries to develop such a best practice based on framework for harmonization of metadata standards.

6 Sharable Content Object Reference Model, see http://www.adlnet.gov/Technologies/scorm/default.aspx

7 As defined in section 2.4

This thesis will show that the term "metadata specification" actually conflates several quite different functions of specifications. It will be argued that interoperability and harmonization will be improved if these functions are more clearly separated into separate components of an harmonization framework.

1.3 Problem Definition

The research questions studied in this thesis concern the definition and application of the terms ”interoperability” and ”harmonization”, and can be summarized in five questions.

1.3.1 Definitions

How can the notions of metadata interoperability and metadata harmonization be meaningfully defined?

Metadata interoperability is seen as a high value ingredient in specifications and systems. While the term ”interoperability” is generally well understood, its application to metadata often conflate very different kinds of issues. A common definition, and a separation between interoperability issues and harmonization issues are necessary to understand the current problems in the field. Section 2 addresses this question.

1.3.2 Measuring Harmonization

What are the features that determine the level of harmonization between metadata standards, and how can they be measured?

Interoperability and harmonization are not zero-or-one quantities – there are different degrees and aspects of interoperability and harmonization. Identifying the features in modern metadata specifications and systems that are central in achieving harmonization is important in order to find the right approaches to improving the harmonization of metadata specifications. By identifying the relevant features, such as extensibility, identification mechanisms etc., and their corresponding quantifiable dimensions, it becomes possible to measure and compare the harmonization of metadata specifications. Section 6 discusses the important harmonization features.

1.3.3 Harmonization Issues

Where does harmonization fail in currently widely used metadata standards?

It will be shown that current metadata specifications suffer from a fundamental lack of harmonization. By using the identified harmonization measures, we can learn more precisely in what ways current metadata specifications fail when it comes to harmonization. The goal is to isolate common problematic design patterns and technologies. In this thesis, it will be argued that many of the issues surrounding interoperability and harmonization are deeply connected to the notion of metadata semantics. Section 6 discusses the harmonization failures in current metadata specifications.

1.3.4 Increasing Harmonization

> *What are the potential methods of increasing harmonization, and how can they be implemented?*

With knowledge of common obstacles for harmonization it is possible to analyze methods and strategies for increasing harmonization between metadata specifications. To be realistic, such strategies must take the concrete environment of metadata specification organizations into account. The goal is to produce concrete guidelines adapted to each metadata specification for taking significant steps toward metadata harmonization. Section 6.5 presents some concrete methods for improving harmonization in current metadata specifications.

1.3.5 Harmonization Framework

> *Can a harmonization framework be formulated that captures the solutions proposed in this thesis?*

Increased harmonization of metadata standards promises to dramatically improve syntactic and semantic metadata interoperability as well as modularity of metadata systems. An attempt is made in this thesis to define a metadata harmonization framework, aimed at providing concrete guidance on increasing harmonization, and adapted to the practical considerations of metadata specification organization as well as to the theoretical harmonization results of this thesis. Section 7 presents such a framework.

1.4 Research Methodology

The research described in this thesis has been performed in close collaboration with the affected metadata communities, with a multitude of practical standardization attempts and standardization developments being part of the collected research data.

Therefore, the analysis is firmly grounded in current needs, motivations and implementations in metadata standardization, and the results can not be seen as mainly theoretical. Rather, the approach chosen is highly applied, focusing on realistic prospects for constructive improvement based on the history and current state of the standards.

The research methodology can therefore properly be described as constructive research, as described by Lukka (2003) and Kasanen et al. (1993). Dodig-Crnkovic (2010) argues that the constructive research method is very common in computer science, although rarely part of the methodological discussion. The constructive approach is described by Dodig-Crnkovic as

> *Constructive research method implies building of an artifact (practical, theoretical or both) that solves a domain specific problem in order to create knowledge about how the problem can be solved (or understood, explained or modeled) in principle. Constructive research gives results which can have both practical and theoretical relevance. The research should solve several related knowledge problems, concerning* feasibility, improvement *and* novelty. *The emphasis should be*

> *on the theoretical relevance of the construct. What are the elements of the solution central to the benefits? How could they be presented in the most condensed form?*

The research presented in this thesis is explicitly directed at building a theoretical model for metadata standardization that helps solve the issues in metadata harmonization. The framework for harmonization presented in section 7 forms the main achievement of the research, and is of both practical and theoretical value.

The results are developed within feasible limits of current metadata standardization practices and are guided towards the improvement of the current standardization process. A number of novel solutions for metadata standards are proposed.

Kasanen et al. (1993) presents the constructive research method using five components:

1. The practical relevance of the research. In this thesis, the practical relevance is demonstrated by the metadata harmonization issues in the current metadata environment that are presented.
2. The theoretical background. In this thesis, the background is formed by current knowledge about metadata, semantics and standardization.
3. The construction of a solution, which forms the main content of the thesis.
4. The practical functioning of the solution. In this thesis, this means demonstrating the practicality of the framework in section 7. Not all parts of the framework have been implemented, but practical future roadmaps are presented for those parts.
5. The theoretical contributions of the research. In this thesis, the theoretical results are a range of analytical tools for describing metadata standards and analyzing harmonization problems.

In this thesis, the research process has consisted of the following elements:

1. Theoretical analysis of current metadata specifications with respect to descriptive features and harmonization issues.
2. Practical experiments in metadata semantics and metadata harmonization, in particular the work on DSP (section 5.5.2), SHAME (Paper 5), and Edutella (Paper 3).
3. Participation in standardization activities aimed at increased harmonization. Section 3 contains a more detailed description of the participation in standardization activities.
4. Publishing research results in forums closely associated with the standardization communities, such as the DCMI conferences (Paper 4 and 5), the WWW community (Paper 1) and the LOM community (Paper 2).
5. Development of a framework for addressing the harmonization, based on the theoretical analysis, the practical experiments and the concrete standardization situations. This has been carried out within the context of the standardization organizations and has been presented in e.g. Paper 2 and 5.
6. Application of the ideas in the framework on the practical metadata standardization developments. See Paper 6.

The process has been highly iterative, where the practical results have been implemented and a new problem analysis made, while the theoretical results have been fed back into the body of theoretical knowledge. The process can therefore be depicted as in Figure 1.1.

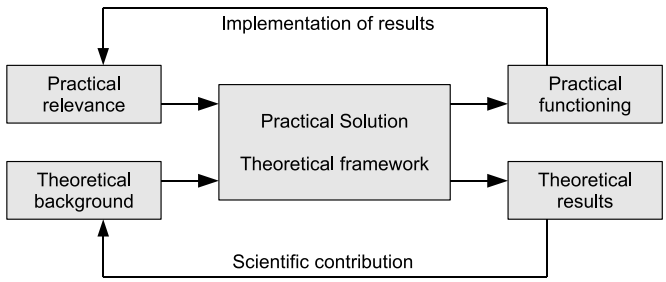


Figure 1.1: The research methodology used in this thesis

Together, this forms a classical example of constructive research in the field of computer science.

1.5 Related work by the author

The author has participated in a number of projects related to metadata that are not part of the main content of this thesis. However, as the results are related to the research described here, a short summary of these projects is included below.

Conzilla

The conceptual browser Conzilla, described in Palmér & Naeve (2005) and first implemented in Nilsson & Palmér (1999), was an early trigger in the KMR research group for requirements for educational metadata interoperability. Conzilla replaces the content-to-content links of regular web browsers with a conceptual browsing interface that supports viewing content through its conceptual context. The effect is a browsable “ontological” view of digital or non-digital content. The applications to collaborative learning (Naeve et al., 2006) and mathematical descriptions (Nilsson, 2002) have further highlighted the need for broad metadata harmonization.

Edutella

The RDF-based peer-to-peer learning object discovery network Edutella, described in Nejdl et al. (2002) and in Paper 3, has been a highly relevant testbed for metadata harmonization, since the network is completely schema-agnostic. It has been very useful for understanding the benefits and challenges of cross-domain harmonization.

Technology-enhanced Mathematics Education

The work of the author on a global infrastructure for content sharing in mathematics education, described in Nilsson & Naeve (2004) and Naeve & Nilsson (2004), has relied heavily on a working distributed, cross-domain metadata infrastructure.

E-learning platforms

Work on a generalized platform for e-learning systems has been described in Naeve et al. (2005) and Palmér et al. (2001). Metadata interoperability and semantics formed part of the envisioned framework, and has later been partly realized through the repository and digital portfolio system SCAM (Palmér et al., 2004).

1.6 Outline of this Thesis

Section 2 lays the groundwork for the rest of the thesis by presenting the concept of metadata and formulating the decisive definitions of interoperability and harmonization.

Section 3 presents the domains of metadata for teaching, learning, libraries and web metadata – the core metadata domains discussed. The relevant metadata specifications discussed in this thesis are introduced. Much of the work behind this thesis has been performed inside several of the relevant standardization organizations. The design of the corresponding specifications rely heavily on the historical relationships between these organizations, and some of that history is therefore also described.

Using the definitions and knowledge of the metadata specifications, an interoperability analysis of metadata syntax and semantics is presented in section 4. A fundamental definition of abstract metadata models is developed.

Section 5 discusses “vertical” harmonization, the internal harmonization within a community centered around a single metadata standard. An important tool for vertical harmonization is application profiles, and a thorough analysis of harmonization aspects of application profiles is presented.

Section 6 provides a detailed analysis of “horizontal” harmonization between independent metadata standards, as compared to metadata crosswalks, and analyzes the necessary components for metadata harmonization. A set of principles for metadata harmonization is presented, based on the notion of semantic embeddability.

In section 7, an evolvable framework for harmonization of metadata standards is introduced, based on the conclusions of the previous sections. The framework is intended to serve as a scaffolding for harmonization, where significant flexibility is combined with far-reaching interoperability. A list of possible steps for the different standards to increase harmonization is identified.

Section 8 summarizes the conclusions of the thesis, and point to possible future directions of research and developments.

2. Metadata and Interoperability

2.1 Background

Metadata as a broad concept is not something new. Library catalogs, as an example, are a form of metadata with a relatively long history. Such catalogs allow librarians to manage a large library without unnecessarily having to deal with the physical books themselves. Geo-spatial information in the form of maps, which allow you to manage land, adding labels and borders without being there, are older still. Or consider gravestones, which give you information about deceased persons and families. In general, metadata is used to refer to all information that describes things.

The two latter examples also highlight the fact that the term "metadata" can be used to include descriptions that provide information about things that are not necessarily information artifacts, but may be, for example, physical entities or even pure conceptualizations such as political borders.

Today, the term “metadata” usually refers to information with one fundamentally different characteristic as compared to these more historic notions: it is *machine-processable*, i.e. it is expressed in a way that allows computers to search, sort and present metadata without human intervention. That is, the “data” in metadata refers specifically to information that is readily accessible to computers. Metadata in this modern sense has been part of computer systems since their early days, for example in file systems where file names and file permissions constitute metadata about the file content. It was in this context the term “metadata” became widely used, in the sense of *data about data* (e.g. Duval 2001, Cabinet Office, 2006), or more explicitly (National Information Standards Organization, 2004)

> *“structured information that describes, explains, locates, or otherwise makes it easier to retrieve, use or manage an information resource”*

With increased computerization in the last couple of decades, metadata has gradually gained a new kind of importance. Geographically separated networked systems with different implementations, but managing the same kinds of data needed to communicate, and metadata standards focused on interoperability between computer systems were developed. Early examples of metadata standards include standards for library information exchange (the MARC format with roots in the 1960s) and standards for geo-spatial data, used for map making (such as NASAs “DIF” format originally from 1987, Directory Interchange Format (DIF) Writer's Guide, (2009)). Another early metadata standard of enormous importance is IETF RFC 822 (Crocker, 1982) from 1982 that specifies the format of e-mail headers, enabling email systems to transfer messages from the sender's computer to that of the addressee.

The growing use of Internet technology and in particular the World Wide Web has become a strong driving force for the development of more generally applicable metadata standards (Baca et al, 2008). With the rise of the web as a platform a whole new usage pattern of metadata has surfaced. Not only are the resources described of a much more diverse nature, but the applications using the metadata are also of many different kinds, and distributed over many more computer systems. The users of metadata are no longer only large, industrial computer systems but also individuals in front of their desktop computers and, more recently, mobile phones and Internet-connected gadgets.

This diversity of systems and resources has led to important new functional requirement on metadata standards in general. In particular, the requirements for cross-domain interoperability are becoming stronger as systems become more and more complex and the amount of information exchange increases. This has been called the third generation of information systems (Sheth, 1999).

2.2 Defining the Metadata Concept

The central characteristic of metadata is its “aboutness” - the fact that something is being described. Therefore, when trying to define metadata, two central questions need to be answered:

1. What kinds of things is metadata about?
2. What are the necessary characteristics of a metadata description?

With carefully developed answers to the above two questions, we can arrive at a definition of the concept of metadata that is useful when discussing interoperability.

2.2.1 What Kinds of Things Is Metadata About?

It can be argued that the definition of metadata mentioned in the previous section, as “data about data”, may be too narrow because it does not allow for information about non-digital things, such as persons, places or books.

In practice, many modern metadata standards go beyond the narrow limitation to “information resources” and instead adopt a definition of metadata that allows descriptions of digital or non-digital things alike, usually collectively termed *resources* or simply *things* (Halpin, 2006) The notion of interoperability is highly relevant for both kinds of information.

A relevant comparison is the definition of “resource” – the subject of metadata descriptions in the context of the Resource Description Framework – inherited from the definition of URIs in RFC 3986 (Berners-Lee et al 2005):

> *the term "resource" is used in a general sense for whatever might be identified by a URI*

If this view is adopted, a thing needs only be identifiable, i.e., distinguishable from all other things, in order to be described by metadata. In short: *we must know what we are talking about*, even though there might be more than one way of actually constructing an identification method. This is a basic requirement for being able to reach a minimum level of interoperability.

For example, when using ID3 tags in an MP3 file, the thing being described is defined implicitly by the context in which the tags appear, and need not be given a URI. This kind of implicit reference to the described thing is commonplace in metadata specifications – it is only assumed that the metadata producer and consumer knows the identity of the thing.

We will later return to a discussion regarding how improved identification conventions lead to increased metadata interoperability.

2.2.2 What Are the Necessary Characteristics of a Metadata Description?

The informal “data about data” definition may be considered too broad because it allows any kind of “descriptive data”, such as an image of a learning object, to be considered metadata, making the metadata concept void of any practical meaning. In this case, the “aboutness” is implicit in a reference accessible by humans, but inaccessible to machine processing, and therefore outside the reach of interoperability considerations.

An important requirement for metadata interoperability is therefore that the metadata is machine processable, and explicitly encoded as metadata. The definition of the data must have an interpretation as being information about a thing.

This is generally achieved in metadata standards by strictly limiting the type of information that is allowed to only a very restricted kind of data, as defined in the data model of the metadata standard, and providing an interpretation of this data in terms of information about the described thing. We will call this characteristic of having a descriptive interpretation *descriptive data*, where “data” implies being machine-processable.

2.2.3 Definition of Metadata

Based on the above consideration, this thesis will use the term “metadata” in the broader sense, not restricted to only information resources. We will also take note of the need for metadata to not only be descriptive, but also processable by computer systems. The following definition summarizes these aspects:

> **Metadata***: Descriptive data about identifiable things.*

This definition encompasses information in at least three broad categories, following the classification in Lambe (2007) (see also National Information Standards Organization, 2004; National Information Standards Organization, 2007):

- "descriptive" metadata, both human-assigned information about a thing (such as name/title, subject and creator), and technical aspects of the thing (size, format, functionality, etc.)
- "administrative" metadata, such as information about the life cycle of a piece of information (different versions, history, etc.)
- "structural" metadata, describing relations between and aggregations of things (such as the relationship between a lesson and its comprising learning objects)

The definition encompasses information not only about digital things, but also about e.g.

- persons and roles, such as learners and teachers (their learning history, competencies, etc.)
- events in time and space (location, participants etc.)
- purely abstract notions (pedagogical designs, terms in taxonomies etc.)

We will later return in more detail to the notion of machine-processability, which is central for understanding the future developments in learning object metadata standards.

2.3 The Notion of Interoperability

What, then, do we mean with the all-important term *interoperability* in a metadata context? IEEE defines interoperability (IEEE Standard Computer Dictionary, 1990) as

> *the ability of two or more systems or components to exchange information and to use the information that has been exchanged.*

That is, interoperability is a characteristic of *computer systems*. The definition is unnecessarily limited, as interoperability can quite reasonably be applied to technical systems outside the field of ICT (such as railway systems or electrical networks). In spite of this, the definition is to a high degree applicable to metadata, being an information-based artifact.

A critical point in the definition is the meaning of "using the information", which in implies using the exchanged data in a way that is consistent with the intentions of the system that created the data. In the case of metadata, this means that the interpretations of the data as descriptions of a thing should be consistent. Metadata created by a human user 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 user who created the metadata.

Applying the definition to metadata therefore results in the following definition:

> **Metadata interoperability**: *the ability of two or more systems or components to exchange descriptive data about things, and to interpret the descriptive data that has been exchanged in a way that is consistent with the interpretation of the creator of the data.*

A central purpose of metadata standards is to contribute to the implementation of interoperable systems. As will be more thoroughly described later, this is generally achieved through the specification of one or more of the following:

1. a common metadata syntax and data formats that aid in consistent parsing of exchanged metadata
2. an abstract model that provides a common framework for the interpretation of metadata
3. a common vocabulary for describing things, that provides shared definitions and interpretations
4. a formal mathematical model for the data, that enables automatic machine inferencing
5. a convention for customizing the standard to a particular system, while retaining interoperability with other systems. Such customizations are often referred to as a "application profiles"

2.4 Metadata Harmonization – Raising the Expectations for Metadata Interoperability

Duval, Hodgins, Sutton, and Weibel (2002) set forth four fundamental principles for metadata 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.

In Nilsson et al (2006a), 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.

We can see that these principles go beyond the requirements of metadata interoperability, as the principles assume a context where multiple metadata standards co-exist. These interoperability concerns therefore not only depend on multiple systems implementing the same specification, but assume a situation where metadata conforming to *different* specifications are used in combination.

In this thesis we will therefore use the term *metadata harmonization* to refer to interoperability in *the presence of multiple metadata standards*. Harmonization can thus be defined as

> **Metadata harmonization**: *the ability of two or more systems or components to exchange combined metadata conforming to two or more metadata specifications, and to interpret the metadata that has been exchanged in a way that is consistent with the intentions of the creators of the metadata.*

Metadata harmonization refers to the ability to correctly process several *different* metadata standards in *combination* within a *single* software system.

On the surface, this definition seems to build on the functionality of software systems. However, by defining metadata harmonization in terms of an invariance between two systems, and by making sure that the metadata interpretation is what's "left" when you factor out the two systems, the above definition of metadata harmonization is actually independent of the systems, and instead describes a feature of the metadata specifications involved. Thus, *metadata harmonization is about the combinability of data.*

An important goal of this thesis is to identify obstacles to harmonization that arise from the design of the metadata standards involved. In that analysis, the five harmonization principles presented above form a useful basis for evaluating metadata harmonization.

It should be noted that the last of the five principles above suggests that given the right support, harmonization may be realized in an automated fashion, with no need for translations, mappings or other manual interventions. Examining this possibility is an overarching theme in this thesis.

2.4.1 Metadata Model Levels

We will analyze the concept of harmonization based on the classification of metadata models developed in Haslhofer & Klas (2010), in turn based on the Meta Object Facility (Object Management Group, 2006), and illustrated in Figure 2.1. In this diagram, metadata specifications are analyzed based on a four-level model – level 0 are the metadata instances, level 1 are the metadata element vocabularies (schemas) and level 2 are the abstract metadata models. Level 3 is the model used to formulate abstract models (in the DCMI case, UML has been used for that purpose).

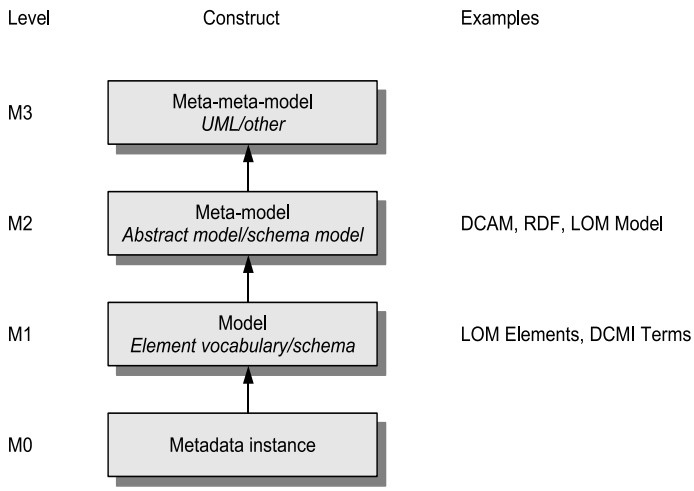


Figure 2.1: Metadata model levels

We can see that the interoperability principles in Duval, Hodgins, Sutton, and Weibel (2002) go beyond the M1 level – they are features of the metadata meta-model. By contrast, Haslhofer & Klas (2010) presents an analysis of interoperability issues on level M1 and M0, assuming harmonization issues on level M2 have already been resolved. By contrast, this thesis will focus exactly on the issue of incompatible meta-models.

2.4.2 Vertical and Horizontal Harmonization

We will analyze two different approaches to improving metadata harmonization. The division is based on a distinction between pre-coordinated harmonization within a controlled set of standards vs. post-coordinated harmonization between independent standards.

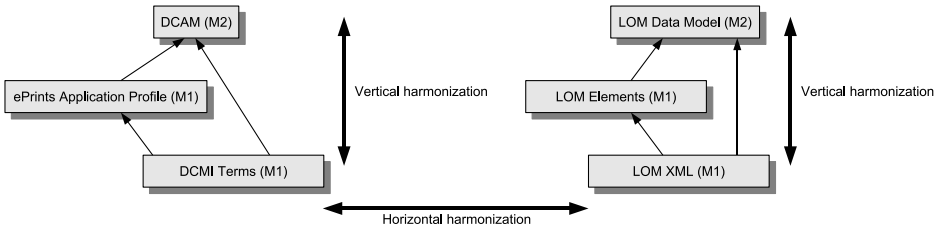


Figure 2.2: Horizontal vs. vertical harmonization

- **Vertical harmonization** – interoperability on different levels within a given set of standards, based on pre-coordination of a base standard.
- **Horizontal harmonization** – interoperability based on interoperability across standards, i.e post-coordination not based on a common standard

Figure 2.2 shows how the two terms compare – vertical harmonization being a concern within the framework of a single model on the M2 level, and is the main focus of the analysis in Haslhofer and Klas (2010), while horizontal harmonization focuses on the relationship between independent metadata standards.

As should be clear from the introduction, the main focus of this thesis is horizontal harmonization, but many achievements in vertical harmonization, such as application profiles, are also interesting design goals for more advanced horizontal harmonization.

3. Metadata Standardization

This thesis builds on research carried out in the context of a number of widely used metadata standards and specifications in the fields of learning, teaching, libraries and multimedia.

While many reserve the word “standard” for technical documentation produced by an accredited international organization such as ISO or IEEE, this thesis will use the terms *metadata standard* and *metadata specification* interchangeably. The reason is that many of the de facto standards in widespread use are specifications produced by other kinds of organizations, such as the World Wide Web Consortium or the Dublin Core Metadata Initiative.

This section presents the metadata standards discussed throughout the thesis, as well as a summary of the contributions and participation of the author in the various standardization initiatives.

3.1 Metadata in the E-learning Domain

The metadata standards discussed in this thesis have all been chosen based on some kind of relevance for the field of e-learning and learning objects.

There are currently a number of metadata standards in use within the e-learning domain. IEEE Learning Object Metadata, published in 2002, is usually regarded as the dominant standard in this field, but in recent years it has become apparent that standards from other communities, such as digital libraries, digital multimedia and e-Government also play an important role for e-learning systems.

The reason is simple: many potential learning objects have their origin in other kinds of repositories of digital content, and their metadata, while described in a way that fits the original community, is of great value in an e-learning context too. Thus, the division of resources into categories with independent metadata standards, such as LOM for “learning objects”, MARC for “library material” etc. is fading in favor of a broader notion of multi-purpose content with multi-purpose metadata.

Apart from the IEEE LOM standard, some of the most important metadata standards that are relevant for learning objects are:

- The Dublin Core set of specifications, popular on the World Wide Web and in the digital library community;
- The Resource Description Framework, RDF, a W3C specification for web-enabled metadata.
- MPEG-7, a complex metadata standard for digital video;
- A set of library related standards: MODS, an XML encoding of parts of the de facto library metadata standard MARC; METS, a metadata container format; and Resource Description and Access, the new library cataloging standard.
- A number of specifications from the IMS Global Learning Consortium, such as IMS Metadata, IMS Content Packaging, IMS Question and Test Interoperability and IMS Learner Information Package that have metadata parts.

Additionally, a number of metadata standards and specifications that are based on one of the above are also relevant. Based on Dublin Core are for example EdNA, a metadata standard for the Australian Education Network, and GEM, a US government-sponsored Gateway to Educational Materials. Based on LOM we find among many others the RDN/LTSN LOM application profile (RLLOMAP) and the Curriculum Online Metadata Schema. The IMS metadata standard and SCORM also reuse LOM as a basis on top of which they build their own frameworks.

These various standards and specifications have been developed to meet different requirements, and to support the needs of different communities. In some cases the standards reflect the broadly shared requirements of a large community; in others, they reflect more specific requirements of a smaller or more specialized community, perhaps defined by activity/interest or by geopolitical boundaries.

The development and usage of these specifications has highlighted the necessity of being able to use component parts of different standards in combination – in other words, the importance of metadata harmonization. Because these standards are not designed to be compatible, they have been a fruitful focus of the harmonization research of this thesis.

3.2 The Dublin Core Set of Specifications

The Dublin Core Metadata Initiative was started in 1995 as a reaction to the problems of finding resources on the rapidly growing World Wide Web. It is used worldwide by a broad range of systems and organizations on the WWW and in various closed infrastructures.

Initially, Dublin Core consisted of 15 metadata terms which were designed to express simple textual information about resources (see Weibel (2009) for an interesting first-hand account of the DCMI history). The project has since grown to accommodate around 80 terms, some of which are of a general nature (such as “title” and “subject”), while others are community-specific (such as “educationalLevel” or “bibliographicCitation”) and still others are used for classification of resources (such as “MovingImage” or “Text”).

The terms in Dublin Core come in three kinds: *properties* (also called "elements"), *syntax encoding schemes* and *vocabulary encoding schemes*. Properties are used to describe a specific aspect of a resource, while the two kinds of encoding schemes are used to specify details of the value of a property. Properties are defined independently of each other, and Dublin Core allows metadata containing any number and combinations of properties to be used to describe a resource.

The term "Simple DC" is sometimes used to describe a usage pattern of Dublin Core metadata that limits itself to the original 15 terms in the Dublin Core Element Set, used in a pattern where each is optional and repeatable.

3.2.1 Relevant Specifications

The core specification is the **DCMI Metadata Terms** (DCMI Usage Board, 2008) document, describing the semantics of the Dublin Core terms, giving each term an universal identifier in the form of a URI and formal relationships to other terms (such as "creator" being a subproperty of "contributor"). The terms are also described in a machine-processable way in accompanying RDF Schema[8] files.

The **DCMI Abstract Model** (Powell et al., 2007) gives the underlying framework for Dublin Core metadata, defining the notions of bounded metadata graphs (*description sets*) properties, syntax encoding schemes, vocabulary encoding schemes etc. The accompanying **DC-TEXT** (Johnston, 2007) format provides a corresponding formal syntax.

Dublin Core metadata can be encoded using one of several syntax specifications, of which **Expressing Dublin Core metadata using the Resource Description Framework (RDF)** (Nilsson et al., 2008a) and **Expressing Dublin Core metadata using HTML/XHTML meta and link elements** (Johnston & Powell, 2008) are current.

3.2.2 Participation

Participation in the DCMI community, with its strong ties to several other metadata communities, notably the library and e-learning communities, has been a central source of interoperability experimentation and development for this thesis.

Following the work on an RDF binding of IEEE LOM, the author participated in finalizing the first version of the **DCMI Abstract Model (DCAM)** in 2005 as well as the second version in 2007 (Powell et al., 2007). During 2009 and 2010, the author has pioneered a radical reformulation of the Abstract Model that makes the model fully harmonized with RDF. This reformulation is yet to become a DCMI Recommendation. The importance of the Abstract Model is described thoroughly in section 4.3 and in Nilsson et al. (2006a).

In parallel to the work on the Abstract Model, the author has led the work on **Expressing Dublin Core metadata using the Resource Description Framework (RDF)** (Nilsson et al., 2008a) through two versions in 2007 and 2008. This has been an important piece in the harmonization efforts between Dublin Core and RDF, together with the new version of the **DCMI Metadata Terms** revision in 2008 that introduced formal semantics for the metadata terms.

8 RDF Schema, or RDF Vocabulary Description Language, is a W3C specification for describing vocabularies designed for use in RDF

In 2009, the author contributed the document **Interoperability Levels for Dublin Core Metadata** (Nilsson, Baker & Johnston, 2009) which describes four degrees of interoperability for metadata applications using Dublin Core. This was a partial outcome of the work of the author on the draft **Description Set Profiles: A constraint language for Dublin Core Application Profiles** (Nilsson, 2008c) from 2008, defining a language for machine-processable definitions of application profiles. The first full definition of the notion of Dublin Core Application profiles was presented by the author in **The Singapore Framework for Dublin Core Application Profiles** (Nilsson, Baker & Johnston, 2008b), also in 2008. This work has been partly documented in Paper 5.

In the context of the **Dublin Core Education Community**, the author has also contributed to the development of new vocabulary and principles for harmonization with IEEE LOM.

3.3 Resource Description Framework

RDF was designed as an extensible framework for metadata descriptions. It was created in 1999, within the Semantic Web initiative at the World Wide Web Consortium (W3C).

The Semantic Web is a visionary project initiated by the W3C with the stated purpose of realizing the idea of having data on the Web defined and linked in such a way that it can be used by machines not just for display purposes, but for automation, integration and reuse of data across various applications.

The Semantic web initiative was motivated by the very same problems that motivate the development of metadata standards: the fact that raw media, in the form of text, HTML, images or video streams, contains meta-information that may be readily deducible from the context for the human consumer (the name of the author, the kind of material contained within, etc.), but which is mostly inaccessible to computers. Making this information available to computers in order to enable a whole new class of semantics-aware applications, was the driving vision that created the Semantic Web project (Berners-Lee et al., 2001).

As described in Paper 1, traditional metadata approaches tend to be based on the assumption that metadata is mostly useful as a digital indexing scheme to use in cataloging and various forms of digital repositories. What distinguishes the Semantic Web from these approaches to metadata are two important things:

- The Semantic Web is designed to allow reasoning and inference capabilities to be added to the pure descriptions. This includes stating simple facts such as "a hex-head bolt is a type of machine bolt", but extends to the inference of new relationships from known data. This is an important feature to allow intelligent agents and other software to not only passively consume descriptions, but to act on them as well.
- The Semantic Web is a web-technology that lives on top of the existing web, by adding machine-readable information without modifying the existing Web. It is designed to be globally distributed with all that this implies in terms of scalability, robustness and flexibility.

The Semantic Web is a layered structure. XML forms the basis, being the standardized transport format. RDF provides the information representation framework, and on top of this layer, schemas and ontologies provide the logical apparatus necessary for the expression of vocabularies and for enabling intelligent processing of information.

This includes the definition of semantic mappings between overlapping metadata standards. As the metadata constructs are based on a common semantic model, the maximal complexity of mappings and the level of precision in mappings are dramatically increased in comparison to mappings between standards using different abstract models (Uschold and Gruninger, 2002).

3.3.1 Relevant Specifications

The core model of RDF is specified in **Resource Description Framework (RDF): Concepts and Abstract Syntax** (Klyne & Carroll, 2004). This model defines the interpretation of RDF metadata and how to construct RDF descriptions, but does not specify a concrete syntax. Accompanying this model is the **RDF/XML Syntax Specification** (Becket, 2004), defining an XML-based expression of RDF metadata. Several other syntax specifications are in widespread use, such as Notation 3 (Berners-Lee & Connolly, 2008) and Turtle (Becket & Berners-Lee, 2008).

The **RDF Vocabulary Description Language 1.0: RDF Schema** (Brickley & Guha, 2004) specification defines how to describe vocabularies for use in RDF metadata, and is itself based on RDF. The formal semantics of RDF and RDF Schema is defined mathematically in **RDF Semantics** (Hayes, 2004). This semantics is the basis for the ontology specifications of the W3C, **OWL Web Ontology Language** from 2004 and the more recent **OWL 2 Web Ontology Language** (from 2009).

3.3.2 Participation

The author has not contributed to the RDF set of specifications.

3.4 IEEE LOM and the IMS Standards

The IEEE LOM standard has its origins in earlier work within the European ARIADNE project and the IMS Global Learning Consortium, beginning in 1995. The first version of the IMS metadata specification was published in 1998, but the development of the standard was eventually transferred to IEEE. In 2002, IEEE finally approved LOM as an international standard, and LOM has since enjoyed an ever-increasing support from other specification bodies and application developers within the e-learning field.

The LOM standard describes LOM-based metadata in terms of a single hierarchy of 76 elements classified into nine categories, and specifies vocabularies and allowed syntaxes for the value of each element. It can be used to convey not only metadata useful for resource discovery, but also information such as aspects of the lifecycle of a learning object and its pedagogical features.

While the terms in Dublin Core are defined and used independently of each other, the LOM standard specifies the structure of the whole of its hierarchy of metadata in a single standard. The standard specifies where in this hierarchy each element may appear, whether it may be repeated, whether ordering matters, and so on. The meaning of a LOM element depends on its precise structural context within a LOM metadata record.

In effect, LOM specifies both the elements themselves and a set of rules for using the elements in combination, a basic example of a so-called "application profile". One advantage of this approach is that it allows for much stricter validation of LOM data as compared to Dublin Core, something that makes LOM immediately usable without further customization.

The IMS Global Learning Consortium[9] has created a diverse set of standards for use in e-learning systems. Although only one of them, the (nowadays) LOM-based IMS Metadata specification, calls itself a metadata standard, there are a number of standards within IMS that, as a whole or in part, fit our definition of a metadata standard. The part of IMS Content Packaging that specifies how to describe the structure of a package of learning objects would classify as a metadata standard, as would the description of a learner in IMS Learner Information Package, etc.

The recent IMS LODE Information for Learning Object Exchange (ILOX)[10] specification solves a similar problem as does the METS specification – the structuring of a set of related metadata descriptions. ILOX is based on FRBR[11] and has been deployed as part of the LRE application profile 4.5[12].

3.4.1 Relevant Specifications

The core standard is **IEEE 1484.12.1, Standard for Learning Object Metadata** that defines an abstract data model of IEEE LOM-based metadata, with the full hierarchical structure and specified data types and vocabularies.

This standard is then implemented in *bindings*, syntactical encodings that conform to the LOM data model. The only standardized binding so far is the **IEEE 1484.12.3, Standard for Learning Technology — Extensible Markup Language (XML) Schema Binding for Learning Object Metadata**, which provides a relatively straightforward mapping from the LOM data model to an XML structure defined by an XML Schema.

3.4.2 Participation

In 2001, the author led the development of an experimental RDF binding of the IMS Metadata specification, published in an appendix to the **IMS Learning Resource Meta-Data XML Binding version 1.2**[13], also discussed in Nilsson (2001a).

9 http://www.imsglobal.org/

10 See http://www.imsglobal.org/LODE/spec/imsLODEv1p0bd.html

11 Functional Requirements for Bibliographic Records, see Tillett (2003)

12 Learning Resource Exchange, a European project of the European Schoolnet (EUN), see http://lre.eun.org/

13 http://www.imsglobal.org/metadata/imsmdv1p2p1/imsmd_bindv1p2p1.html

Starting in 2002, the author lead the development of **IEEE P1484.12.4 Standard for Resource Description Framework (RDF) binding for Learning Object Metadata data model**, described in Paper 2. The project eventually led to the conclusion that a straightforward binding of IEEE LOM to RDF was unrealistic due to modeling difficulties[14]. Instead, in 2005 the draft binding was withdrawn and replaced by two standardization projects, also led by the author:

The **IEEE P1484.12.5 Standard for Resource Description Framework (RDF) Vocabulary for IEEE Learning Object Metadata (LOM) Data Elements** has the goal of producing a standardized RDF vocabulary capturing the metadata properties built into the LOM data model in RDF compatible expressions.

The **IEEE P1484.12.4 Recommended Practice for Expressing IEEE Learning Object Metadata Instances Using the Dublin Core Abstract Model** is designed to complement the LOM RDF vocabulary standard. It uses the definitions of metadata terms defined by the LOM RDF vocabulary standard together with DCMI metadata terms for expressing IEEE LOM conforming instances as description sets conforming to the Dublin Core abstract model.

The above two harmonization standards, in late draft versions at the time of writing, are in development in the **Joint DCMI/IEEE LTSC Taskforce**[15] led by the author, with the goal of presenting the two documents for ratification by both communities. The work within this taskforce on the above two documents provides an important foundation for the analysis in this thesis.

3.5 The Library Metadata Standards: MODS, METS, RDA

In the field of library metadata, the dominant standard for bibliographic information has long been the arcane MARC format, with roots in the 1960s. The MARC standard with its peculiarities and not very machine-friendly format is unsuitable as a basis for metadata harmonization (Coyle & Hillmann, 2007). Several approaches for addressing this problem have been developed.

MARC-XML[16] is a direct XML translation of MARC designed as a stepping stone between MARC and other metadata formats. It retains all of the MARC structure, semantics and data types, but uses an XML syntax. While it improves dramatically on the machine processability of the MARC format, many of the core problems of MARC metadata still remain.

The Metadata Object Description Schema (MODS)[17] format has been developed by the Library of Congress to serve as a modern version of the MARC format. It is designed using XML technology and conventions, which makes it a more interesting object for harmonization efforts. MODS can be used in conjunction with the Metadata Encoding and Transmission Standard (METS)[18], which essentially is an XML-based container format for bibliographic metadata, designed to provide metadata about bibliographic records in a multitude of formats.

However, the most interesting development in the library domain is most certainly the Resource Description and Access (RDA) standard (see Coyle & Hillmann, 2007), published in 2010.

14 See further discussion in section 6.2.3

15 http://dublincore.org/educationwiki/DCMIIEEELTSCTaskforce

16 http://www.loc.gov/standards/marcxml//

17 http://www.loc.gov/standards/mods/

18 http://www.loc.gov/standards/mets/

It is designed as a replacement for the comprehensive cataloging guidelines known as the Anglo-American Cataloging Rules (AACR2)[19], with origins in the 19th century. The main purpose of RDA is to provide detailed rules for identifying, transcribing and structuring bibliographic metadata. Building on the conceptual bibliographic model in the Functional Requirements for Bibliographic Records (FRBR)[20], it is not designed as a concrete metadata format, even though many view it as the foundation for a future MARC replacement.

RDA defines, in abstract terms, a set of metadata elements together with relevant vocabularies for elements such as Content Type. An important issue for RDA has been to ensure that the definition of these elements are future-proof, so that they can be reused in other metadata specifications such as Dublin Core. Therefore, since 2007, an effort to describe the RDA metadata elements and vocabularies using the RDF Schema has been in development in collaboration with DCMI.

3.5.1 Relevant Specifications

This thesis will briefly discuss the **Metadata Object Description Schema (MODS)** standard in the context of XML-based metadata format. Though of great historical interest, we will not consider the MARC21 or MARC-XML standards, and the METS standard, while interesting, is too peripheral for the discussions in the thesis.

Though the development of RDA vocabularies expressed in RDF Schema unfortunately has not yet resulted in a formal standardization activity, the process of reinterpretation of RDA in terms of RDF properties and classes is highly relevant for the harmonization discussions in this thesis.

3.5.2 Participation

The author has contributed to the initiation of the development of RDF-compatible expressions of the RDA elements at a meeting between representatives of the Joint Steering Committee for Development of RDA (JSC) and the DCMI held at the British Library in May of 2007, as well as to the work that has been carried out in the context of the DCMI/RDA Task Group that resulted from the meeting. This work has resulted in draft RDF vocabularies in December 2009 (Hillmann et al., 2010).

3.6 ISO MLR

When IEEE LOM was first standardized in 2002, it was also submitted to ISO for a so-called “fast track” standardization, which implies that an external standard is submitted in completed format for ratification without substantial changes.

19 http://www.aacr2.org/

20 See Carlyle (2006), IFLA Study Group (1998) and http://www.ifla.org/publications/functional-requirements-for-bibliographic-records

The submission was handled by the ISO/IEC JTC1 SC36 committee for “Information technology for learning, education and training”. It soon became apparent that the fast track process would fail, since significant changes were proposed to the base IEEE LOM standard that were unacceptable to the IEEE. If the proposed changes had been accepted, the result would have been a version of LOM that is incompatible with the IEEE LOM version.

Instead, the SC36 committee decided to initiate a new standardization effort designed as a replacement for IEEE LOM, with the stated goal of addressing many of the identified deficiencies in IEEE LOM, while retaining a level of interoperability with IEEE LOM and, additionally, increasing the interoperability with Dublin Core metadata. In short, the standardization effort was created to increase metadata harmonization in the field of learning technology.

3.6.1 Relevant Specifications

This new multipart standard was given the name **ISO/IEC 19788 Metadata for Learning Resources**. The core of the standard (parts 1 and 2 below) are now in the late stages of circulation, and a final version is expected within a year or two. The following parts are under preparation:

1. Framework
2. Dublin Core elements in MLR
3. MLR basic application profile
4. Technical elements
5. Educational elements
6. Availability, distribution, and intellectual property elements

3.6.2 Participation

MLR has a relatively stormy history. Apart from the initial turbulence as it became apparent that IEEE LOM could not be fast-tracked, the initial developments of the standard was criticized for a lack of connection to the metadata community. It was also criticized for a lack of concern for metadata harmonization in a submission from the DCMI in 2006, formulated by the author in Nilsson (2006b).

Two years later, a relatively far developed draft, called CD3 and published in 2008, received heavy criticism from several of the participating countries for being too long, too complicated, and for adopting a structure-orientated XML approach more oriented toward e-business applications, rather than one more compatible with RDF and Dublin Core. The hierarchical approach had originally been chosen over a standard based on ISO/IEC 11179 (Metadata Registry (MDR) standard), based on many of the participants' previous experience of XML-based standards, but the lack of RDF compatibility now became a serious issue for the standard.

The criticism was summarized in a submission by the author and a list of experts from participating countries titled **Requirements for ISO MLR interoperability** (). The submission was met with approval from the committee, and the signatories were tasked with staking out a new direc-

tion for the MLR standard based on the submission. The result of that process is a heavily updated version of the MLR standard that builds on a semantic model more closely aligned with RDF and Dublin Core, and which is currently being circulated for comments.

We will later return to some remaining harmonization issues with the proposed standard.

3.7 MPEG-7

MPEG-7 (ISO/IEC IS 15938-2:2002) is the name of a digital video standard with a heavy focus on the use of metadata to describe the content of a video stream. What makes MPEG-7 interesting is the fact that it has the potential to be deeply integrated into the video production process, something that generally can be expected to result in very high metadata quality. MPEG-7 also represents a challenge in that the resources it describes can be extremely intangible, such as an appearance of a certain person in a movie. By contrast, other metadata standards such as LOM and Dublin Core have been developed in a library tradition, using a document metaphor.

This metadata standard does not contain any information specific to learning, but several parts of the information embedded in MPEG-7 metadata might still be useful for an e-learning application.

MPEG-7 is also special in that it defines its own, relatively complex, so-called Description Definition Language (DDL) that is used to customize the metadata format to a certain application. Most other XML-based metadata standards rely on XML-based specifications such as XML Schema for the definition of the metadata format.

3.7.1 Relevant Specifications

MPEG-7 was standardized beginning in 2002 in **ISO/IEC 15938 Multimedia content description interface**.

3.7.2 Participation

The author has participated in the discussion in the W3C Multimedia Semantics Incubator Group, which produced its final report in July 2007 (Hausenblas, 2007).

4. Metadata Syntax and Semantics

The basis for metadata is descriptive data and its interpretation. The most obvious harmonization issue is the plethora of metadata syntaxes that are not immediately compatible. It is necessary to understand in detail what role formats play in metadata harmonization, and when syntax is secondary.

Underlying the more superficial syntax incompatibilities are sometimes deeper issues connected to the modeling conventions used when specifying the metadata. A deeper understanding of the modeling issues is central in the development of approaches to metadata harmonization. A definition of abstract models is developed.

Semantics is a complex concept that plays a pivotal role for metadata and harmonization. The notions of metadata semantics and ontologies are handled fundamentally differently in the various metadata domains considered in this thesis. Semantics is a complex concept that plays a pivotal role for metadata and harmonization.

This section will analyze the metadata harmonization aspects of metadata syntax and semantics, based on the discussion in Nilsson et al. (2006a).

4.1 Metadata Model Categories

The metadata standards discussed in section 3 fall into three broad categories:

1. Standards based on an resource – property – value model. These standards use a variant of entity-relationship or graph-based modeling, with clear boundaries between the nodes (often called “resources” or just “things”), and relationships between things. A distin-

guishing feature of these standards is that nodes in the graph represent described things, creating a link between the metadata structure and its semantics. This category includes RDF, Dublin Core and nowadays also ISO MLR

2. Standards based on an abstract hierarchical model. In such models, metadata are clustered inside other metadata elements, with no clear division between resources and relationships. The hierarchy represents the structure of the data, but has no direct relation to the metadata semantics. These include IEEE LOM, the various IMS standards and partially RDA

3. Custom XML languages. Such languages are also hierarchically organized, but expressed using XML terminology, and often rely on XML idioms such as XML Schema. These include MODS, METS and MPEG-7.

From a metadata harmonization perspective we would want to be able to combine information from several different standards in descriptions of the persons, artifacts, events, etc. that make up an e-learning system. In practice, this is currently difficult or impossible to do. Instead, each standard lives in isolation, largely incompatible with the others. The reason for this is not tied to any single standard, but originates in the lack of a common platform for metadata standards in general.

The structure of the XML-based standards, MPEG-7, MODS and METS are in many ways similar to LOM and the IMS standards in that they are complex, monolithic hierarchies of data elements with strict structural constraints, even though the details of how the hierarchies are constructed differ substantially.

In the author's experience, the standards based on abstract hierarchies are often designed with an XML expression in mind. For example, the early versions of the IMS metadata standard were explicitly modeled in XML at the work group meetings (and the abstract version then extracted from the XML format), and XML expressions of the IMS standards have always been published in parallel with the abstract models. In the IEEE LOM case, the XML binding was the first binding to be published, and follows the hierarchy closely.

The two categories also have similar characteristics on a theoretical level. Thus, for the purposes of this section, we will for the most part treat the XML-based standards and the abstract hierarchical standards as a single category.

The RDA case is more difficult. The original RDA element structure was very similar in structure to IMS and LOM metadata, but through the work of the DCMI/RDA Task Group, the element structure has gradually evolved into something more closely resembling a resource – property – value model[21]. As the element structure of RDA is not part of RDA proper, the actual status of the RDA model is still somewhat muddy, as evidenced by Hillmann et al. (2010), and the transition to an entity-relationship model is not complete. RDA can therefore be said to be falling into something of a middle ground, where lessons from both categories of standards may be applicable.

Because of the conceptual similarities between the different standards, it is possible to derive generalizable harmonization results from a comparative study of a smaller set of standards.

21 See http://www.rda-jsc.org/docs/5rda-elementanalysisrev3.pdf

Therefore, the rest of this chapter will focus mainly on examples and analysis based on Dublin Core, LOM and MODS, as this will highlight the most important difficulties with trying to combine two different approaches to defining metadata. However, the lessons learned will be applicable to a much broader range of standards, including the standards mentioned above. IEEE LOM and Dublin Core have been chosen based on the author's extensive experience with standardization within these two communities, while MODS is a canonical example of an XML-based metadata standard.

4.2 Metadata Formats and Extensibility

At a superficial glance, the major problems of metadata harmonization seem to relate to formats: Most standards use incompatible methods of encoding their information, creating difficulties for consuming applications.

The formats currently used by LOM, Dublin Core and MODS actually all allow for extending the format and combining terms from external sources. The problem instead lies on another level, in the *interpretation* or *semantics* of the metadata expressions. In particular, metadata applications will have trouble understanding LOM terms in a DC context, MODS terms in a LOM context, etc.

In order to understand these difficulties, we must first see how the standards tend to approach the issue of metadata formats.

4.2.1 Bindings

Both LOM and Dublin Core use a two-layered approach to defining metadata models. In the core standards, an abstract information structure is defined, defining the terms that may be used and their relationships. This information structure can then be encoded in one of several alternative formats, called *bindings*. As an example, Dublin Core currently supports two bindings[22]:

- "meta" tags in HTML/XHTML
- RDF, the Resource Description Framework, a general-purpose metadata framework

The situation with LOM is similar. An XML binding for LOM was approved by the IEEE in 2005, while a form of RDF binding is in development.

Bindings to other formats than the officially standardized are sometimes necessary, of which some see wide-spread use and others are only used for internal purposes. Many applications use such "private bindings" for, e.g., implementing their metadata in a relational database, or embedding metadata in a private protocol. One such example is the News Metadata Framework[23], which uses a custom version of Dublin Core metadata.

On the other hand, no alternative encodings are available for MODS, as it specified directly in terms of the XML syntax.

22 An older XML binding has been withdrawn, and while a replacement exists in draft form, no version of an XML binding currently has Recommendation status.

23 News Metadata Framework Requirements specification. http://www.iptc.org/dev

It is interesting to note that the bindings discussed here – RDF, HTML and XML – are all specified by the W3C, which should not be surprising as many of the harmonization problems we are studying arise in a WWW context.

HTML "meta" tags will not be further considered in this thesis, due to their limited generality, and the fact that specifications such as RDFa[24] are gradually replacing metadata harmonizations efforts based on meta tags. Instead, we will concentrate on the two major current metadata formats: XML and RDF.

4.2.2 XML-based Formats

An XML document can be represented as a tree structure of *XML elements*. Each element may contain text as well as other XML elements, and may also have *attributes*. While XML has its origins in standards for creating structured markup in text documents, it is widely used to encode data of many kinds.

XML itself does not provide a fixed set of element names and attribute names. Rather, users of XML define their own *XML language*, or in other words: a set of element names and attribute names for use in XML documents and a set of rules for how those named elements and attributes are to be interpreted. For this reason, the XML standard itself is sometimes referred to as a *meta-language*, i.e., a set of rules for defining XML languages.

Thus, an XML language is defined by a syntax plus an accompanying definition of the semantics of the language that is used to extract meaning from the XML structures. Not all such semantics are metadata semantics. Examples of semantics that is not a metadata semantics are XHTML or OpenDocument format[25], where in both cases the interpretation of the syntax is a document rather than a metadata description, or SOAP, where the interpretation is a message intended for remote method invocation.

However, there are also a number of XML languages that fall under the definition of a metadata standard. RSS[26] has an interpretation as information about a news item, or the Sitemap[27] format, designed to convey information about web sites to search engines.

Each of the XML-based metadata standards we discuss in this thesis define their own such XML language. One such language is the LOM XML binding defined by the IEEE, exemplified by the metadata record in Example 4.1.

24 http://www.w3.org/TR/rdfa-syntax/

25 Also known as ODF. See http://opendocument.xml.org/

26 Really Simple Syndication, see http://en.wikipedia.org/wiki/RSS

27 See http://www.sitemaps.org/

```
<?xml version="1.0"?>
<lom xmlns="http://ltsc.ieee.org/xsd/LOM" >
  <general>
    <identifier>
      <catalog>URI</catalog>
      <entry>http://www.example.com/objects/Para101</entry>
    </identifier>
    <language>fr</language>
    <description>
      <string language="en">
        This learning object explains parachuting.
      </string>
    </description>
    <structure>
      <source>LOMV1.0</source>
      <value>atomic</value>
    </structure>
  </general>

  <educational>
    <description>
      <string language="en">
        Useful for learning some flight-related French terminology.
      </string>
      <string language="sv">
       Användbar för att lära sig lite flygrelaterad fransk terminologi.
      </string>
    </description>
    <language>en</language>
  </educational>
</lom>
```

Example 4.1. A LOM XML metadata instance

This XML file is a metadata description of a learning object about parachuting. The LOM XML binding tells us in detail how to interpret each XML element in terms of the LOM data model, which in turn gives us the interpretation of the metadata. In the above example, we can see that although the learning object, which has an “atomic” structure) is in French (“fr”), it is intended for English-speaking learners (“en”), and the real purpose is to learn flight-related French terminology.

The LOM XML binding thus specifies the precise interpretation of each XML element, *in the context it appears*. The interpretation is formulated in terms of LOM elements, LOM categories etc.

As we can see from the example above, the XML element “language”, when taken on its own, is ambiguous; it must be interpreted differently when it appears as a sub-element (or *child*) of the “general” and “educational” elements, respectively. It is therefore necessary for the LOM XML binding to specify the interpretation of the complete XML document as a whole, taking all parent/child relations between metadata elements into account. A single XML element can be mapped to different LOM elements depending on context.

Other XML languages that reuse IEEE LOM metadata are perfectly possible. These will have their own rules for interpreting the XML data, and will operate independently of the official binding. Note that such alternative languages may reuse XML element names from the official bindings, but use them together with a different set of rules. A simple example would be a LOM RSS module[28]

Another XML metadata language is specified in the MODS guidelines. Example 4.2 shows a resource described using MODS and encoded in that language:

```
<?xml version='1.0' encoding='UTF-8' ?>

<mods xmlns:xlink="http://www.w3.org/1999/xlink" version="3.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.loc.gov/mods/v3"
xsi:schemaLocation="http://www.loc.gov/mods/v3
http://www.loc.gov/standards/mods/v3/mods-3-0.xsd">
  <titleInfo>
    <title>Sound and fury :</title>
    <subTitle>the making of the punditocracy /</subTitle>
  </titleInfo>

  <originInfo>
    <place>
      <placeTerm authority="marccountry"
                 type="code">nyu</placeTerm>
    </place>
    <place>
      <placeTerm type="text">Ithaca, N.Y</placeTerm>
    </place>
    <publisher>Cornell University Press</publisher>
    <dateIssued>c1999</dateIssued>
    <dateIssued encoding="marc">1999</dateIssued>
    <issuance>monographic</issuance>
  </originInfo>

  <language>
    <languageTerm authority="iso639-2b"
                  type="code">eng</languageTerm>
  </language>

  <subject authority="lcsh">
    <topic>Journalism</topic>
    <topic>Political aspects</topic>
    <geographic>United States.</geographic>
  </subject>

</mods>
```

Example 4.2. A MODS metadata instance.

From this description and the semantics defined by MODS, we can understand that the resource described is about Journalism (as defined in the Library of Congress Subject Headings), is in English and was published as a monograph by Cornell University Press in 1999, etc. As MODS is not based on bindings, the interpretation as metadata is defined directly in terms of the XML syntax.

28 One such module by Stephen Downes can be found at http://www.downes.ca/xml/rss_lom.htm

4.2.3 RDF

Unlike XML, RDF is not a meta-language, i.e., each specification based on RDF will not create its own incompatible RDF-based language. Instead, RDF is a single framework which allows descriptions using parts from different metadata standards and terms from independent vocabularies to coexist within the *same* metadata instance. It is thus fair to say that RDF has been designed to fulfill the role of a general-purpose metadata language.

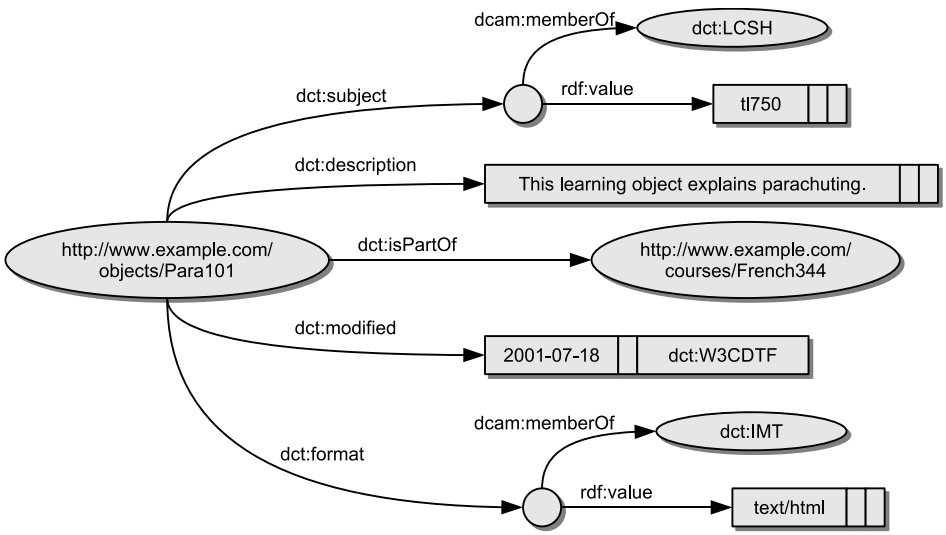


Figure 4.1: An example of a Dublin Core description expressed in RDF.

Also unlike XML, RDF is not specified in terms of a concrete syntax, but in terms of an abstract structure, which is often represented as graphs.

Much like XML, though, RDF has no built-in names, but rely on independent vocabularies to create metadata instances.

RDF metadata is made up of sets of *statements*. Each statement describes a single attribute, or *property*, of a single resource. By combining several statements about the same resource, a metadata description of that resource can be constructed. RDF data can be represented as a nodes-and-arcs diagram, where the nodes represent resources, and arcs represent properties. An example Dublin Core metadata record expressed in RDF is seen in Figure 4.1.

Expressing the LOM example using the draft LOM RDF vocabulary gives us the RDF metadata depicted in Figure 4.2.

In this example, we can see that terms from several standards are combined in a single RDF description. RDF itself specifies a base vocabulary that is used for specifying resource types (the property with the identifier rdf:type), Dublin Core specifies a resource type that is used to represent languages (dct:RFC1766), and LOM specifies a property to be used to describe a resource using a value of that type (lom:educational_language). We can also note that the LOM RDF expression has chosen to reuse Dublin Core properties for expressing common properties such as “language” and “description”.

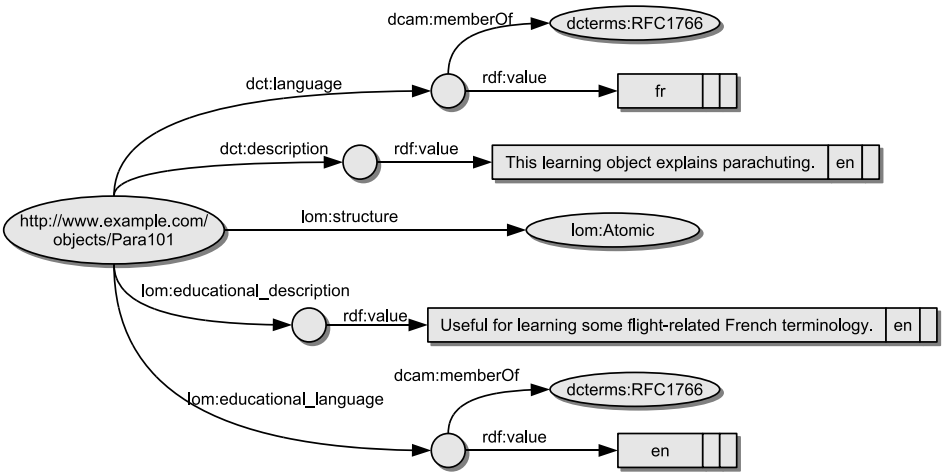


Figure 4.2: An example of a LOM instance expressed in RDF.

While the graph notation for RDF is very useful, it cannot be used for exchanging metadata between computer systems. For this purpose, a serialization of RDF into an RDF-specific XML language can be used. This RDF/XML language is an example of an XML language that may contain XML elements with identical names as XML elements in the LOM XML language (such as lom:description). But as noted earlier, these elements will now be interpreted using the rules of the RDF/XML language instead of the LOM XML language.

It is important not to confuse this RDF/XML serialization with RDF itself, which is not bound to a specific syntax and actually has a multitude of different concrete syntaxes, including several incompatible XML serializations.

It is also important to realize that RDF does not allow for multiple incompatible and context-dependent usages of the same term. In contrast to XML, which allows the reuse of identical XML elements across many different XML languages, with different structural constraints and different interpretation, RDF does not leave room for private semantics of properties. For example, the LOM RDF property lom:language must be used in accordance with the RDF semantics and RDF constraints defined by the LOM RDF vocabulary in all RDF metadata instances. An RDF statement involving this property has exactly the same interpretation independent of context. The

ambiguity allowed in LOM XML, where lom:language is used in two places with two different meanings, must be resolved, for example by introducing a new property, lom:educational_language for carrying the second meaning.

4.2.4 Extending and Combining Metadata Descriptions

We have seen how metadata can be expressed in both XML and in RDF. But can we combine terms from several standards in a single document? The answer is: it depends.

We will use the term *metadata fragment* to mean an interpretable syntactical part of a metadata instance, containing enough of the structure of the metadata instance to have a meaningful interpretation as metadata. In RDF, this means a set of triples (but not just a URI or a literal), while in LOM it means a LOM element with its substructure (but not just a LangString or Vocabulary value).

On the surface it seems straightforward to add metadata fragments from, for example, MODS to a LOM XML document. The specifications even explicitly mention this possibility. Let us say we want to use the educational description from LOM, and the subject from MODS. Example 4.3 is the result of extending a LOM XML document with a fragment from MODS.

```
<?xml version = "1.0"?>
<lom  xmlns="http://ltsc.ieee.org/xsd/LOM"
      xmlns:dc="http://purl.org/dc/elements/1.1/"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

  <general>
    <identifier>
      <catalog>URI</catalog>
      <entry>http://www.example.com/objects/Para101</entry>
    </identifier>

<!-- MODS fragment: -->

   <subject authority="lcsh">
      <topic>Parachuting</topic>
   </subject>

<!-- End MODS fragment -->

  </general>

  <educational>
   <description>
     <string language="en">
       Useful for learning some flight-related French terminology.
     </string>
     <string language="sv">
       Användbar för att lära sig lite flygrelaterad fransk terminologi.
     </string>
   </description>
   <language>en</language>
  </educational>


</lom>
```

Example 4.3. A LOM XML metadata instance, extended with a MODS metadata fragment

As we can see, the MODS fragment describing the subject of a resource can be added into the LOM XML document. Where to place it is flexible – we have chosen a placement inside the <general> LOM category, but LOM allows extensions on all levels of the schema.

On the other hand, we can do the reverse, starting from the MODS XML document and adding the LOM fragment from the element “Educational.Description”. The result is shown in Example 4.4.

```
<?xml version='1.0' encoding='UTF-8' ?>
<mods xmlns:lom="http://ltsc.ieee.org/xsd/LOM"
      xmlns:xlink="http://www.w3.org/1999/xlink" version="3.0"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xmlns="http://www.loc.gov/mods/v3"
      xsi:schemaLocation="http://www.loc.gov/mods/v3
                          http://www.loc.gov/standards/mods/v3/mods-3-0.xsd">

 <subject authority="lcsh">
    <topic>Parachuting</topic>
 </subject>

 <extension>

  <!-- LOM fragment: -->

   <lom:description>
    <lom:string lom:language="en">
      Useful for learning some flight-related French terminology.
    </lom:string>
    <lom:string language="sv">
       Användbar för att lära sig lite flygrelaterad fransk terminologi.
     </lom:string>
  </lom:description>

  <!-- End LOM fragment. -->

 </extension>
</mods>
```

Example 4.4. A MODS metadata description, extended with a LOM XML metadata fragment

In contrast to the MODS-in-LOM example in Example 4.3, the LOM structure needs to be wrapped inside the <extension> MODS element, where all non-MODS structures must be placed. Also note how the LOM element “description” is ambiguous: it can be interpreted either as the General.Description element or as the Educational.Description element, since the relevant LOM context is missing.

How about doing the same kind of combination in RDF? It is just as straightforward: we can merge parts the two diagrams in our RDF examples, and arrive at an RDF description looking like Figure 4.3. In fact, our original LOM RDF example in Figure 4.2 already showcases this kind of combination.

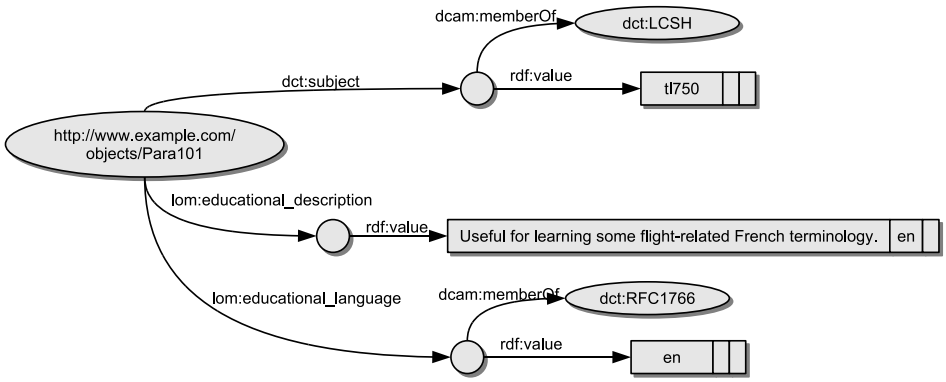


Figure 4.3: A combined LOM and Dublin Core metadata description, expressed in RDF.

One obvious and important difference between RDF and XML is that XML creates two cases: one case where a LOM XML instance is extended with MODS metadata, and one case where a MODS description is extended with LOM XML metadata (and this combinatorial problem increases if we add a third standard to the mix). By contrast, RDF does not distinguish between the two cases – the results are identical.

Mixing standards based on their syntactic representation thus seems possible in both XML and RDF. Unfortunately, straightforward as both examples appear, complex problems start to appear as we examine how metadata applications are to process the metadata we have constructed. The tool we need to understand the difficulties is called *abstract models* and *semantics*, and we now turn to a description of these subject before returning to our examples in section 6.2.

4.3 Abstract Models for Metadata

To take the step from raw data to metadata, a metadata specification must, besides the syntax specification, also define an interpretation of the syntax in terms of information about a thing. This essentially means that the standard must define a mapping from the concrete syntax to some form of meaning of the metadata.

Such an interpretation is a kind of *semantics*, a term which in this context should be understood in a relatively general sense. There are examples of formal metadata semantics using the mathematics of model theory (notably, the RDF semantics in Hayes (2004)), but informal metadata semantics formulated using ordinary language are more common. Section 4.4 will discuss the notion of semantics more thoroughly.

4.3.1 Using Abstract Syntaxes to Define Metadata Semantics

The standards we study in this thesis essentially use two kinds of approaches to defining a metadata semantics. In order to be format-independent, LOM, RDF and Dublin Core all base their semantics on an abstract data structure, or *abstract syntax*, specific to the respective standard. This structure specifies the concepts used in the standard, and how they combine to form a metadata description, but it does not define a concrete syntax or file format that can be used to exchange metadata, nor does it define the meaning of the concepts.

When exchanging metadata using a standard based on an abstract syntax, a piece of information about a resource, such as “this learning object is useful for learning some flight-related French terminology” is first expressed in the abstract syntax, and then encoded using a concrete syntax, such as the LOM XML instance in Example 4.1. As we have seen, such syntaxes are called bindings in the context of LOM.

When a receiving application tries to interpret this metadata, it uses the rules of the LOM XML language to convert the concrete syntax to the abstract syntax. It can infer that “educational” is a LOM category, and that the “string” XML element represents a “string” item within a LOM LangString data type, used as value for the LOM “description” element.

The LOM standard then tells us how to interpret this abstract information, and that the interpretation is that this is a learning object in French for English speaking students,that is useful for learning some flight-related French terminology.

The Dublin Core abstract syntax is similarly used by Dublin Core-based applications as an intermediate layer between the application and the bindings. This fundamental process of *expression/interpretation* is described in Figure 4.4.

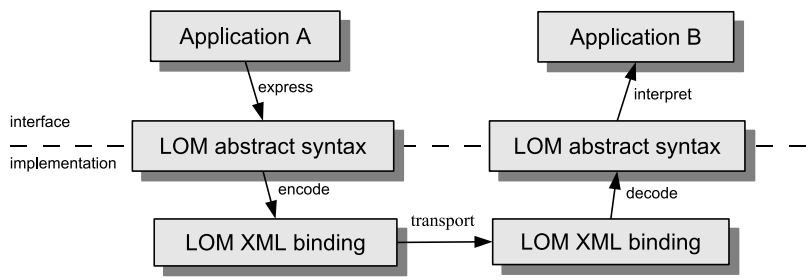


Figure 4.4: The process of encoding/interpretation of metadata

The MODS example shows us that using an abstract syntax is not a requirement for metadata. MODS does not define its own abstract data structure, but instead adopts the concrete syntax of XML[29], and bases its semantics directly on the XML elements and attributes. The same can be said of the MPEG-7 standard.

29 The XML InfoSet is an attempt at formalizing the XML model in a syntax-independent fashion, and can be viewed as an abstract syntax for XML.

4.3.2 Interpreting Metadata Through the Lens of an Abstract Model

As can be seen from both the IEEE LOM and Dublin Core specification documents, the abstract syntax tends to be specified alongside its semantics. Because an abstract syntax for metadata is useless without its semantics, we will use the term *abstract model* to denote a semantics that is based on an abstract syntax for the metadata standard:

> ***Abstract metadata model:*** *A mapping from an abstract syntax to an interpretation of the syntax as information about a thing.*

Note that the definition of such a mapping implies the specification of the domain of the mapping, i.e. the abstract syntax. An abstract model therefore per definition requires the definition of an abstract syntax.

When two applications want to exchange metadata using an abstract model-based metadata standard, they therefore understand the metadata through the lens of the abstract model. The abstract model functions as an opaque interface, an API, to the metadata. In practice, the exchange is realized using one of the bindings, but the details of the formats are of no interest to the applications, which instead analyze the metadata in terms of the interface and interpretation given by the abstract model.

The abstract model is thus the key used by a metadata application to unlock the secrets of a metadata expression given in a specific format, 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.

Because of this, abstract models are essential in understanding metadata harmonization issues. The abstract models of hierarchical metadata standards such as LOM and entity-relationship-based models such as Dublin Core or RDF are fundamentally different in several ways, and these differences are a major source of difficulties when trying to combine the standards. As we will see, applications will find that terms from one standard make little sense if interpreted in the context of the other standard.

Similarly, metadata standards lacking an abstract model, instead being defined directly in terms of a concrete syntax, will face significant harmonization issues when being combined with incompatible abstract model-based standards, not to mention incompatible syntaxes. We will return to these concrete harmonization issues in section 6.

4.3.3 The Dublin Core Abstract Model

An early effort to produce an abstract framework for Dublin Core was presented in Bearman, Miller, Rust, Trant and Weibel (1999). The current Dublin Core Abstract Model (Powell, Nilsson, Naeve and Johnston, 2007) defines the kinds of terms that can be used in Dublin Core metadata descriptions and an abstract syntax that ties them together. The interpretation of the terms is based on RDF.

Just as in RDF, a *property*, identified using a *Property URI*, is used to describe a single aspect of a resource, also identified using a URI, the *Resource URI*. In a Dublin Core metadata description, any number of properties and their associated *values* may be used to describe a resource. The

abstract model tells us that values can be referenced using a *value URI*, and further described in *related descriptions*. Values (such as the names of creators, textual descriptions, etc.) can be represented as *value strings*.

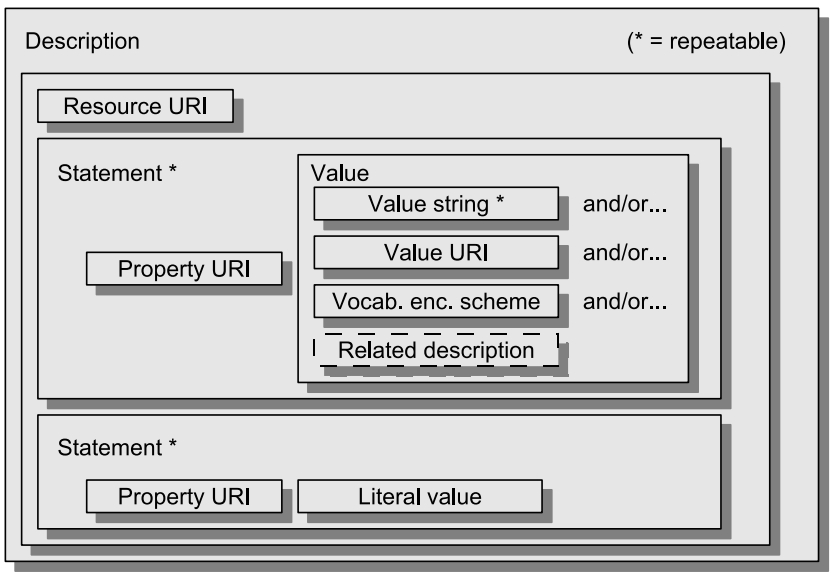


Figure 4.5: A simplified overview of the Dublin Core abstract model.

Syntax encoding schemes can be used to specify the precise syntax of value strings, while *vocabulary encoding schemes* are used to indicate a controlled vocabulary used as source of a value. An overview of the Dublin Core abstract model is found in Figure 4.5.

Using these relatively simple building blocks, it is possible to create very complex metadata descriptions, for example based on the FRBR-based[30] Scholarly Works Application Profile (SWAP/ePrints AP) described in Allinson, Johnston & Powell (2007), which uses a five-entity model to describe the relationships between a scholarly work, expressions, manifestations, copies and their various contributors in a single Dublin Core metadata record.

While some Dublin Core syntaxes do not support all constructs in the abstract model (for example, HTML meta tags do not currently support the notion of vocabulary encoding schemes), the different formats all share the same common understanding of the basic notions of *properties* and *values*.

The Dublin Core semantics is therefore consistent across various syntaxes, and in all cases dependent on the identification of properties, values etc in the data structures. Because of this, a basic interpretation of Dublin Core metadata in terms of entities and their relationships can be

30 FRBR, Functional Requirements for Bibliographic Records, is a specification for the structure of metadata for library usages, and uses a relatively complex five-entity model to describe, e.g., a book. See Tillett (2003)

specified without reference to the concrete elements used, using only the abstract syntax. With added knowledge about the actual element definitions, this basic interpretation can be filled with various kinds of meaning.

This is very much in line with the abstract model of RDF, which maps RDF triples to a basic interpretation in terms of entities and relationships, which can be supplemented using knowledge about the terms used. In fact, there is currently work in progress within DCMI to replace the DCMI abstract syntax with an abstract syntax building directly on the RDF abstract syntax.

4.3.4 The LOM Abstract Model

Similarly, the LOM abstract model uses an abstract syntax to specify the structure of LOM metadata instances. In contrast to the *property-value* structure used by Dublin Core, LOM uses a hierarchical structure of *elements-within-elements*. Each element can be either a container element, thus containing other elements, or a leaf element, which holds a value of a certain data type. The top-level elements are called *categories*.

The abstract syntax of LOM, as seen in Figure 4.6, is somewhat similar to the XML element structure (though the two should not be confused). Unlike XML, LOM does not allow attributes on elements, nor does it allow text content within elements for other than leaf elements.

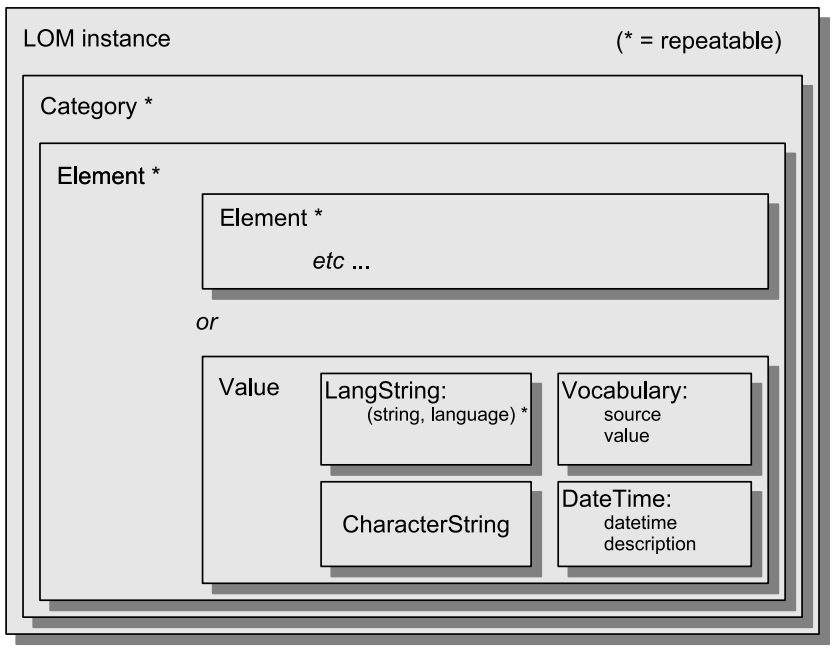


Figure 4.6: An overview of the LOM abstract syntax.

As we have seen, an interpretation of a LOM metadata instance needs to take the element context into account. For example, the element “language” means different things depending on whether it occurs in the context of the General category or the Educational category.

Another kind of ambiguity in LOM is that different element describe different things. Some elements are interpreted as attributes of the learning object, while some (in section 3, Metametadata) are interpreted as attributes of the metadata description, while still others (in section 7, Relation) are interpreted as attributes of a related learning object.

Therefore, the LOM semantics cannot be formulated in general terms, based only on the abstract syntax, but needs to take the concrete LOM elements used in the metadata into account to make any sense at all of the metadata.

This means, on the other hand, that LOM extensions completely lack interpretation in the LOM abstract model and can only be managed as black boxes. This feature is a fundamental obstacle to metadata harmonization in the case of LOM, an issue which we will return to.

4.4 Metadata Semantics

Semantics is the study of meaning, and in the context of computers, semantics is typically used to denote the intended effects a computer program is supposed to perform when processing a given syntax. For example, the intended execution effects of some code in a programming language, or the intended results of an API call.

In the context of metadata, the semantics is defined in terms of the resulting *description of a thing* rather than any specific action or side effect. Any potential side effects of metadata descriptions are, in other words, out of scope for metadata semantics. Metadata semantics thus turns an otherwise meaningless data structure into a description.

Metadata semantics is often designed for human consumption, but how do we handle semantics for machine consumption in metadata standards? It is touched upon in the definition of metadata interoperability and harmonization, which refer to the processing and interpretation of exchanged data.

We can therefore distinguish different kinds of semantics, based on their intended uses:

- **Informal semantics** means all the human semantics that is not accessible to machines, and is generally expressed in plain text in metadata specifications.
- **Machine-processable semantics** means a specification of metadata semantics expressed in a machine-parseable format. Such a format provides avenues for automatic discovery of the meaning of metadata expressions, thus allowing metadata applications to partially understand metadata extensions encountered in previously unknown application profiles.

- **Formal semantics** means a specification of metadata semantics in terms of a formal mathematical model. Such a model provide the foundation for processing metadata in software agents and ontology-based reasoning systems, which in turn provide the basis on which to build machine-processable mappings between semantically overlapping standards. Formal semantic models are generally also accompanied by a machine-processable format.

The following sections will explain these concepts in more detail.

4.4.1 The Role of Refinements in Dublin Core and LOM

The Dublin Core abstract model provides two basic primitives for the machine-processable expression of metadata semantics: sub-properties and sub-classes, adopted from RDF Schema. Both primitives are used to specify so-called *refinements*, that serve the important purpose of allowing more fine-grained descriptions to be understood by applications that only know how to process more coarse-grained descriptions.

Suppose we declare the property "ex:illustrator" to be a sub-property of the Dublin Core element "dct:contributor". Applications that know the difference between "dct:contributor" and "ex:illustrator" may use the values of the two properties in subtly different ways that are appropriate to the situation. However, an application that does not know how to process the "ex:illustrator" property may still choose to process the value of that property in the exact same way that it would process a value of the "dct:contributor" property. Thus, a resource with an "ex:illustrator" of "Gary Chalk" may be said to simultaneously have an implicit "dct:contributor" of "Gary Chalk". The formal word for this process of implicit and automatic "creation" of property values is *entailment.*

Note that the process of entailment is mandatory in the sense that it is considered invalid to specify a value of the "ex:illustrator" property that is not at the same time a valid value for "dct:contributor". This must of course be reflected in the definition of the sub-property: if not all valid values of the sub-property are also valid values of the property, the sub-property definition is invalid. For example, while the values of an "ex:owner" property are sometimes also valid values of "dct:contributor" (as owners sometimes also participate in the creation of a resource), this is not *always* the case. Thus, "ex:owner" cannot be declared a sub-property of "dct:contributor". The details of how to define refinements and some of their consequences are given in Johnston (2005b).

The other kind of refinement, sub-classes, is used together with the specification of the type of a resource using the "dct:type" property. For example, the type "dctype:StillImage" is a sub-class of "dctype:Image". Sub-classing simply means that everything that is of the type "dctype:StillImage" is simultaneously of the type "dctype:Image". This allows for a fine-grained specification of resource types, while allowing for interoperability with less capable applications.

The process of simplifying metadata records based on refinements is sometimes referred to as *dumb-down*, as it can be used to construct a less refined, but more widely processable metadata record. It can be performed by the application itself, or in a pre-processing step.

LOM does not have a corresponding notion of refinement. In fact, the LOM standard states that due to interoperability concerns, "extended data elements should not replace data elements in the LOM structure". And, in fact, a contributing reason for this is that there is no machine-process-

able way to specify that a LOM extension refines a LOM element. Therefore, an application would not be able to recognize that an extended LOM element can be processed in the same way as the LOM element it replaces, or dumbed-down to the original LOM element.

4.4.2 Formal and Informal Semantics

Returning again to our metadata format examples, let us try to understand how an application arrives at an understanding of metadata expressions.

When processing the LOM XML example in Example 4.1, an application will first need to know what XML language is being used, as the XML document itself generally does not specify that information. So, given that we know that our data is given in the LOM XML format, the interpretation of each XML element is given by the LOM XML binding – a "description" XML element within an "educational" element must be interpreted as the "5. 10 Description" LOM element in the LOM category called "5. Educational". The LOM standard itself specifies the human semantics of this element: "Comments on how this learning object is to be used".

Note that in this process, the interpretation must be performed by reference to the published LOM standards. Any machine processing must be manually tailored to each and every element of the metadata structure. This is an example of *informal* semantics, or semantics that is explicit, but not machine-processable.

Let us contrast the previous example with the RDF example from Dublin Core in Figure 4.1. An RDF application will process the RDF metadata and find an RDF property named "dct:format". An application can use the URI of the property to obtain a description of the property provided by the authority that defines it (the Dublin Core Metadata Initiative), using the RDF Schema language. That description includes human-readable information about the property, and also machine-processable data describing its relationships to other resources, including refinement relationships with other properties.

The value of the property, "text/html", is seen by the application to be an member of the vocabulary "dct:IMT". The Dublin Core RDF Schema provides human-readable information to indicate that this vocabulary is the set of all Internet Media Types, or MIME types; it also provides machine-processable data describing the relationship of this class to other resources.

The fact that "dct:format" is a property and "text/html" is an member of the vocabulary "dct:IMT", and further information based on the descriptions of that property and that class, can be inferred with no human intervention.

What we find here is an example of *machine-processable* semantics, where an application can automatically process the metadata structure to arrive at a partial understanding of the metadata. If the metadata includes properties that refine other properties, these refinements can also be processed automatically, for example in order to perform a dumb-down of the metadata record.

Note that the application does not need to know what metadata standard it is processing, but only needs access to the corresponding machine-processable RDF schemas that describe the element and value vocabularies used in the description. This points to a major difference between XML-based languages and RDF: XML-based languages provide their own, often incompatible semantics. XML specifications such as XML Schema are limited to capturing syntactic features of

XML languages, and cannot describe their semantics. It is therefore a reasonable conclusion that XML-based metadata standards such as MODS or MPEG-7 that allow unrestricted XML constructs, will necessarily be limited to informal semantics.

Similar conclusions can be drawn regarding IEEE LOM. As we have noted, the LOM abstract model lacks important semantic information, such as regarding what parts of the LOM structure are about what thing. A LOM extension basically lacks semantics from the point of view of a LOM consumer, leaving LOM interoperability at the purely syntactic and informal semantics levels. A machine-processable semantics for LOM would require significant modification to the LOM abstract model to be realized, even though it might not be completely impossible to design.

On the other hand, RDF provides a basic framework for metadata semantics that all standards expressed in RDF conform to, based on the RDF abstract syntax. The formal semantics of RDF is specified in Hayes (2004), and basic semantics of RDF metadata terms can be expressed using the RDF schema language (Brickley and Guha, 2004). Dublin Core has chosen to use RDF Schema as a way to express the formal, machine-processable semantics of the Dublin Core properties and encoding schemes, for use also in metadata formats other than RDF.

Not all machine-processable semantics are based on a *formal* mathematical model. ISO MLR is an example of a metadata standards that defines a machine-processable semantics (though there is yet no specified syntax for it), but fails to provide a formal model for the semantics. We will soon return to this issue in the context of ontologies below.

An interesting discussion of different kinds of metadata semantics can be found in Uschold and Gruninger (2002). The approach to metadata found in the RDF set of standards has many intriguing features that might serve as a source of inspiration for future learning object metadata standards, so we now turn to a short introduction to RDF and the Semantic Web.

4.4.3 RDF and the Semantic Web

RDF has been created to enable the vision of the “Semantic Web” – a web of machine-processable information, extending the current web. RDF tries to reach this goal by:

- Using a coherent framework based on URIs for identification of metadata elements such as properties, classes and resources. RDF is perhaps best described as a “semantizisable” web, which provides a sufficiently coherent metadata framework that its component parts can be given proper formal semantics without inconsistencies or ambiguities.
- providing a basic abstract model for metadata, with certain built-in semantics. This basic model allows applications to store and process metadata from different standards in a common framework.
- being extensible, both structurally and semantically. We have already seen examples of semantic extensions in the form of refinements, as well as proof of the straightforwardness of structural extensions when combining several metadata standards.
- being web-capable, unlike traditional databases and knowledge representation systems. While the RDF model is based on previous work on knowledge representation systems, it differs substantially in that it integrates with WWW standards such as XML and URIs.

- being decoupled from the information it describes, rather than closely tied to the data it describes the data. In RDF, anyone can express any statements about any resource. It is up to the application to determine trustworthy sources. This allows for multiple descriptions, appropriate for different contexts, of a single resource to co-exist.
- allowing for self-describing metadata. Thanks to its machine semantics, RDF applications can partially process new metadata without previous knowledge of the standards involved.

The RDF standard (Klyne and Carroll, 2004, Manola and Miller, 2004) is by its very nature a semantic standard. In RDF, the tokens used in the format do not merely identify syntactic elements, but by design refer to notions in the real world. By contrast, XML elements are by themselves only syntactic placeholders that need the semantics of an XML language to be given meaning (Cover, 1998). Similarly, the statements expressed in RDF are not just data structures, such as is the case with XML document trees, but have real-world meanings. Every RDF statement has a real-world interpretation, independently of any other RDF statement. RDF can therefore be described as *a framework for extension and recombination of independent statements about things.*

4.4.4 Vocabularies, RDF Schemas and Ontologies

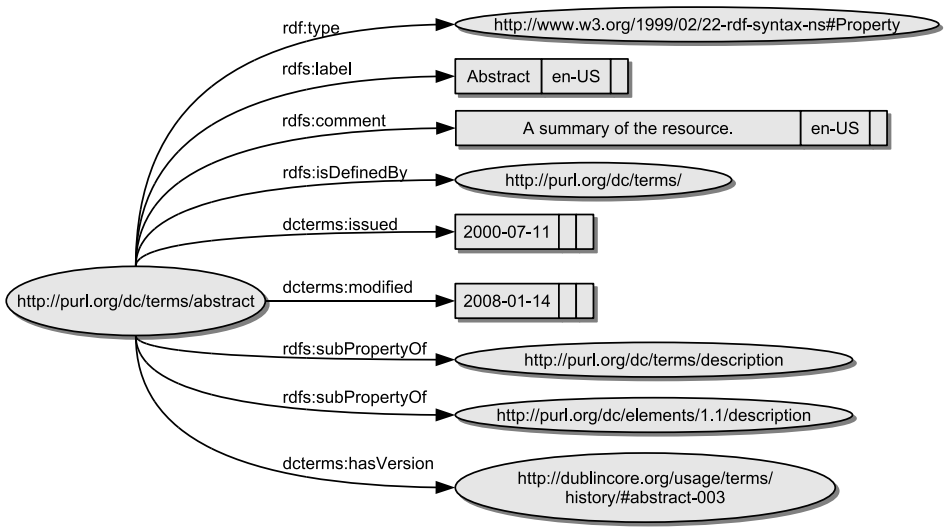


Figure 4.7: The RDF schema description of the Dublin Core term "dct:abstract".

Using RDF Schema, parts of the semantics and properties of terms can be expressed in a common framework. For example, Dublin Core provides one set of terms, and the LOM RDF binding provides another. RDF schema allows for the description of relationships between terms not

only within one single standard, but also across standards. It also allows for description of any number of attributes of the vocabulary terms themselves, using any RDF properties. For example, the Dublin Core term "dct:abstract" is described by the Dublin Core RDF schema as depicted in Figure 4.7. These kinds of descriptions of metadata terms aid in the interpretation of metadata, and can therefore be seen a form of machine-processable semantics.

RDF Schema contains a base semantics that is used in practically all RDF descriptions, and that encompasses both property refinement and sub-classing. The following table gives some examples of what can be expressed in RDF Schema, and using what construct.

In order to express	*Use this construct*
This resource is a Person	rdf:type
Student is **a kind of** Person	rdfs:subClassOf
"creator" is a **Property**	rdf:Property
"hasBirthday" **can only be used to describe** a Person	rdfs:domain

Another promising RDF-based framework for defining RDF terms, especially in the form of hierarchical taxonomies or thesauri is SKOS, Simple Knowledge Organization System (Miles and Brickley, 2005).

For more advanced semantics, *ontologies* using the Web Ontology Language OWL, provide a foundation for expressing complete conceptual models of a domain, allowing for a dramatically higher level of automation that allows computer systems to operate at a conceptual level much closer to the human level. As described in Heflin (2004), OWL can express that the Person and Car classes are disjoint, or that a string quartet has exactly four musicians as members, something that RDF Schema cannot do.

Another important benefit of ontologies is that they allow for the automatic deduction of additional information about resources based on existing information. For example, if the metadata of a certain learning object states that it requires support for a specific set of standards, such as CSS2 and XHTML, and it is separately known which web browsers support those standards, an inference engine can infer that a certain browser works with that learning object without being explicitly told so. In the same way, ontologies provide support for semantic mappings between vocabularies that partially overlap, so that users may ask questions in terms of one vocabulary and receive answers that are described using a separate vocabulary.

The use of ontologies requires a formal mathematical underpinning of the metadata model, as ontologies need to be defined with mathematical precision. The traceability of ontology calculations also depend on formal expressions of the metadata semantics. Therefore, very few metadata frameworks are in a position to support ontologies as few are based on a formal model – of the most widely used standards, only RDF is[31].

31 This rules out for example ISO Topic Maps (ISO/IEC 13250), which lacks a formal model, and formal ontology-based description languages such as KIF (Knowledge Interchange Format) not in widespread use.

A more thorough discussion of ontologies is beyond the scope of this thesis, which deals with more fundamental issues in metadata harmonization. It should be clear from the short description above that the use of ontologies presumes far-reaching metadata harmonization, or alternatively, the use of a single metadata standard only.

4.4.5 Semantic Metadata Interoperability – a Cornerstone for Harmonization?

Shared semantics is, by definition, a necessary feature of metadata interoperability, and is therefore a central feature of all metadata specifications.

The above discussions show that the RDF family of specifications are special in one particular sense; not only do they use a shared informal semantics for human consumption, but they also enable machine-processable semantics. That is – an important part of the interpretation of the metadata is expressed in an explicit, formal form using schemas and ontologies usable for machine processing.

Therefore, systems that implement the semantics of RDF can achieve *interoperability of their metadata semantics*. We use the term *semantic metadata interoperability* to capture a situation where two systems can exchange machine-processable semantics alongside the metadata and interpret this semantics correctly.

Semantic metadata interoperability has potentially very important consequences for metadata harmonization, where the central problem is ensuring metadata is interpreted consistently across various contexts – both in combination with other metadata and across systems.

We therefore put forward the following hypothesis as a major possible conclusion of this thesis:

> **Hypothesis:** Semantic metadata interoperability is a precondition for practical metadata harmonization.

The following sections will address this hypothesis from several perspectives.

4.4.6 Interoperable Processing and Ad-hoc Processing

Even for standards supporting semantic metadata interoperability, it is certainly fully possible to produce applications that process metadata without regard to the machine semantics. An example would be an XSL transform that extracts specific information directly from the Dublin Core in RDF/XML syntax. Such *ad-hoc processing* of metadata records requires that the precise content of the records is well-known in advance. For example, such an application cannot process a Dublin Core metadata record that includes a refinement of an element. An application trying to use the syntactic content of the XML element “dct:contributor” will not be able to process a metadata record that uses “dct:creator” instead, even though the latter implies the former.

In contrast, *interoperable processing* is based on the abstract model and the interoperable semantics, and is necessary when an application needs to be prepared for metadata constructs that do not fall within the limits of a limited syntactic description. Interoperable processing does not use the metadata syntax directly, but relies on the higher level interface provided by the abstract model, and processes metadata with knowledge about the semantics.

As metadata interoperability requires a full understanding (within the scope of the metadata specification) on the part of the metadata consumer of the intentions of the metadata producer, it should be clear that interoperable processing is a basic prerequisite for metadata harmonization in the context of machine-processable semantics.

Metadata standards not based on an abstract model (such as the XML-based standards), or not using machine-processable semantics (such as IEEE LOM), rely on direct processing of the syntax and are therefore not subject to this distinction.

4.5 Summary

Based on Nilsson (2010) and the above discussion, we can summarize the structure and models of common metadata standards in the following table.

Specification	*Structure*	*Syntax*	*Syntactic extensions*	*Semantics*
IEEE LOM	Hierarchical	Abstract	Additions to the tree at any point	Informal
The DCMI specifications	Entity-relationship	Abstract	Any conforming term can be used at any point	Formal
RDF	Entity-relationship	Abstract	Any conforming term can be used at any point	Formal
ISO MLR	Entity-relationship	Abstract	Any conforming term can be used at any point	Machine-processable
RDA	Hybrid tree-based and entity-relationship	Abstract	Not defined	Informal
MODS	XML tree	XML	XML Schema extensions	Informal
MPEG-7	XML tree	XML	XML Schema and DDL (Description Definition Language) extensions	Informal

5. Vertical Harmonization

This section focuses on vertical harmonization, which can be defined as harmonization designed to ensure that systems implementing a base standard or a set of base standards are interoperable regardless of what kind of implementation of the standard the systems choose. The underlying assumption is that there is more than one way of implementing the standard.

This includes considerations about how a standard enables harmonization with extensions of the standard, as well as adaptations of the standard using application profiles. Application profiles, designed to combine, restrain or extend metadata standards, are a central tool in vertical harmonization. The conventions differ substantially between different metadata specification traditions, and will there be given special consideration in this section. A thorough analysis of the general problems associated with vertical harmonization, with a focus on translating between element vocabularies can be found in Haslhofer & Klas (2010).

We give examples of vertical harmonization from IEEE LOM, Dublin Core and RDF.

5.1 Vertical Harmonization in IEEE LOM

In LOM, there are two dimensions of vertical harmonization: conformance levels and syntax bindings.

LOM defines two conformance levels in the base LOM standard:

- **Strictly conforming** LOM metadata instances, meaning metadata that consist only of LOM data elements, i.e., extensions are not allowed
- **Conforming** LOM metadata instance, meaning metadata that may contain extensions.

This points to two kinds of application profiles: *restricting profiles* that only add additional constraints to the base LOM standard and therefore remain within the limits of strictly conforming instances; and *extending profiles* that additionally may add new metadata elements and therefore will not guarantee strict conformance. A LOM-consuming application will need to decide which

of these two conformance levels it supports, leading to different levels of harmonization with regards to various LOM profiles. We will return to LOM application profiles in section 5.5.3 below.

Syntax bindings of LOM offer an additional dimension of vertical harmonization, thanks to the LOM abstract syntax which allows applications to be interoperable on the syntax-independent level or to depend on a particular syntax.

5.2 Dublin Core Interoperability Levels

In 2009, DCMI published a document called “Interoperability levels for Dublin Core Metadata” describing four “levels of metadata interoperability” (Nilsson, Baker & Johnston, 2009) for metadata applications and specifications that wanted to use Dublin Core. These levels are a good description of the vertical harmonization dimensions for Dublin Core metadata.

The purpose of the document is to help application developers and metadata designers in understanding that there are more than one way for a metadata specification to interoperate with other Dublin Core implementations. The four levels describe the “choices, costs, and benefits” involved in aiming for a certain kind of interoperability. The levels are designed as a ladder, where higher-level interoperability build on the lower levels. The levels are

1. **Shared term definitions.** On this level, only the natural language definitions of the Dublin Core terms are reused. This is the level on which the Dublin Core ISO standard operates. On this level, systems will not be interoperable as there is no technical standard involved, but the human interpretation of the metadata will be guided by a common set of term definitions.
2. **Formal semantic interoperability.** On this level, the formal definitions of the Dublin Core terms as RDF properties and classes are reused. This is the level on which RDF-based applications operate. On this level, interoperability is based on the RDF interoperability mechanisms, such as URI-based identification, merging of metadata descriptions and interpretation of RDF Schemas.
3. **Description Set syntactic interoperability.** On this level, the Dublin Core notion of Description Sets[32] for defining metadata records is used. This is the level on which applications and specifications based on the Dublin Core abstract model operate. Dublin Core-specific syntaxes and abbreviations can be used interoperably.
4. **Description Set Profile Interoperability.** On this level, metadata specifications and applications use the DSP model to specify and validate metadata records. This is the level on which Dublin Core Application profiles as defined by the Singapore Framework[33] operate. On this level, a high level of interoperability is achieved, even on the level of the complete structure and content of a metadata record.

32 See section 5.5.2

33 See section 5.5.2

5.3 Vertical Harmonization in RDF

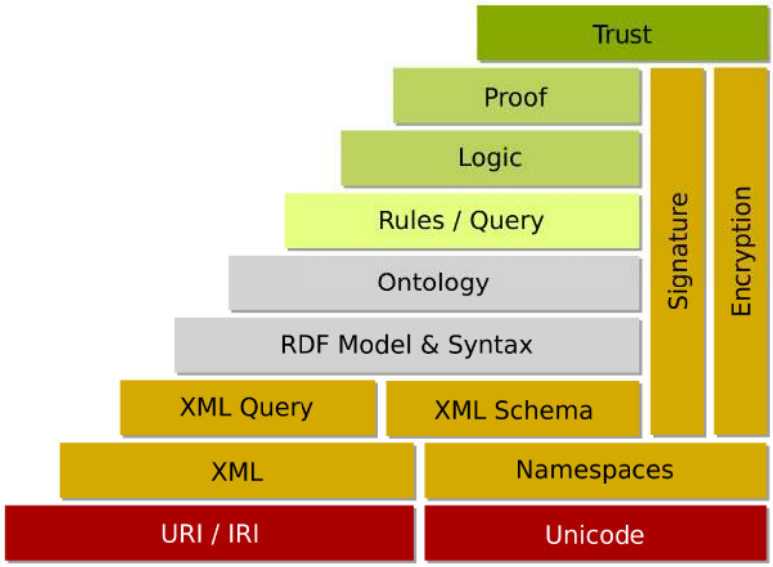


Figure 5.1: The Semantic web layered model

In RDF and the Semantic Web, vertical harmonization is integrated with Web architecture, and is often presented using a layered model as in Figure 5.1.

This model defines interoperability in terms of the stack of supporting specifications, starting with Unicode and URIs, through XML and XML namespaces, RDF and going all the way to ontologies (OWL, a W3C Recommendation), rules (RIF, currently Proposed Recommendation) and trust (no specification yet). Though XML cannot really be seen as part of the RDF framework, it is true that RDF is grounded in Web Architecture.

5.4 Vertical Harmonization in XML-based Metadata Specifications

For XML-based specifications, the LOM pattern of two levels is common: strict conformance vs. support for extensions, even though the precise details of the nature of extensions may differ substantially. MPEG-7 stands out from the rest thanks to its additional layer, the Description Definition Language, an extension to XML Schema that is a requirement for full MPEG-7 interoperability.

Some XML languages are designed for reuse within a family of specifications. For example, the IMS Content packaging standard includes the XML-based IMS Metadata by reference. In these cases, standards tend to be reused as complete XML trees rather than by reusing individual elements, necessary due to the context-dependent nature of XML elements and document-like characteristics of XML.

5.5 Application Profiles

In order to support community-specific and regional needs, many metadata standards support a notion of customization through *application profiles*. Enabling such customizations of metadata standards is one of the ultimate goals in the process of improving metadata harmonization as we have described it in this thesis, and for application profiles, harmonization between metadata standards matters in a very concrete way. In this section we will describe how application profiles rely on the harmonization capabilities of the respective metadata standards, and how application profiles still live in the realm of vertical harmonization.

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 hurdles for the very harmonization issues that application profiles have been designed to solve.

These different approaches to application profiles depend, to a large extent, on the differences in abstract models. Therefore, solving the abstract model issues paves the way for a harmonized approach to application profiles, with significant improvements in metadata harmonization as a result.

As much of the focus in harmonization discussions historically has been directed at application profiles, we will describe their background in some detail.

5.5.1 Metadata Standards and Profiling

The community that develops and uses a metadata standard is rarely completely homogeneous. It is common that in order to be useful to a community of reasonable size, a metadata standard incorporates some degree of flexibility. The developers of services that make use of that standard take advantage of this flexibility to customize the standard to meet the specific requirements of their service and its audience.

In some cases, such customization may involve selecting some subset of the full descriptive capability provided by a rich or expressive metadata standard, on the basis that not all of the functions supported by the standard are required in the context of a particular service. In other cases it may involve enhancing the specificity of description to support some particular requirements of a targeted user community.

The term *profile* has been widely used to refer to a document that describes how standards or specifications are deployed to support the requirements of a particular application, function, community or context, and the term *metadata application profile* has been applied over the last decade to describe this tailoring of metadata standards by their implementers.

The process of “profiling” a standard introduces the prospect of a tension between meeting the demands for efficiency, specificity and localization within the context of a community or service on the one hand, and maintaining interoperability between communities and services on the other. Furthermore, different metadata standards may provide different levels of flexibility: some standards may be quite prescriptive and leave relatively few options for customization; others may present a broad range of optional features which demand a considerable degree of selection and tailoring for implementation.

We also noted earlier that the development of the World Wide Web has had an impact on the use of metadata and on the development of metadata standards. One effect of this changed environment is the development of metadata standards that are designed to support generic functions and to be applicable to a broad range of types of resource: the Dublin Core is an example of such a standard.

Another perhaps more subtle aspect is a growing recognition that it is desirable to be able to use community- or domain-specific metadata standards – or component parts of those standards – in combination. It should not be necessary to perform complex, costly and sometimes incomplete mapping of metadata each time resources or metadata move across community boundaries, particularly since, as noted above, new mappings must be designed each time a new community with a different standard joins the network of communication partners.

Rather, it is argued, the implementers of metadata standards should be able to assemble the components that they require for some particular set of functions - and if that means drawing on components that are specified within different metadata standards, that should be possible – safe in the knowledge that the assembled whole can be interpreted correctly by independently designed applications. Duval et al (2002) employ the metaphor of the Lego set to describe this process: an application designer should be able to “snap together” selected “building blocks” drawn from the “kits” provided by different metadata standards to build the construction that meets their requirements, even if the kits that provide those blocks were created quite independently.

Another motivating factor in this approach is the pragmatic desire on the part of the developers of metadata applications to make use of existing work and reduce redundant duplication of effort. If an implementer of metadata standard A has developed a component - say, a classification scheme or controlled vocabulary - which another implementer using metadata standard B regards as useful within their application, they should be able to “reuse” that existing component easily. And further, applications processing the metadata descriptions from the two sources should be able to establish that those reused terms are indeed the same terms.

Heery and Patel (2000) present a compelling vision of metadata implementers “mixing and matching” “data elements”, constructing application profiles by selecting from the sets of “data elements” provided by metadata standards and by other implementers. Hillmann & Phipps (2007) show how application profiles are a potentially powerful tool for machine validation of metadata and evaluation of metadata quality.

In the cases of both the Dublin Core and LOM metadata standards, standards developers and implementers recognize the application profile as a mechanism for realizing the goals of metadata modularity, extensibility and refinement. Both communities have developed some guidance for the creation of such application profiles, which offer at least some measure of the mixing and matching capability outlined by Heery and Patel (2000). See also “Dublin Core Application Profile Guidelines” (2003), Baker (2003), Duval and Hodgins (2003) and IMS Global Learning Consortium (2000).

As has been argued, the extent to which the DC and LOM standards meet their ambitious goals of extensibility and modularity, and the form in which that extensibility and modularity are implemented, is determined by features of the different abstract models underlying the standards. And indeed this fundamental dependency is reflected in the fact that the two communities present different approaches to the metadata application profile. In both cases, an application profile enumerates the set of terms that may be referenced in some set of metadata descriptions, and provides some, perhaps context-specific, information about how those terms are to be used. Beneath that general similarity, however, lie some significant differences.

5.5.2 Dublin Core Application Profiles

In a Dublin Core application profile, the terms referenced are, as one would expect, terms of the type described by the Dublin Core Abstract Model, i.e. a Dublin Core application profile describes, for some class of metadata descriptions, which properties are referenced in statements and how the use of those properties may be constrained by, for example, specifying the use of vocabulary and syntax encoding schemes. The DC notion of the application profile imposes no limitations on whether those properties or encoding schemes are defined and managed by DCMI or by some agency: the key requirement is that the terms referred to in a DC application profile are compatible with the DC Abstract Model.

It is a condition of that abstract model that all references to terms in a DC metadata description are made in the form of URIs. The URI is a global identifier system. As long as the owner of a URI adopts policies which guarantee the persistence of the URIs they assign - i.e. they provide assurances that once a URI is assigned to a metadata term, it will continue to identify that metadata term and will not be used for another resource - the requirement for unambiguous identification of terms is met. Terms can be drawn from any source, and references to those terms can be made without ambiguity.

This set of terms can be regarded as the “vocabulary” of the application or community that the application profile is designed to support. The terms within that vocabulary may also be deployed within the vocabularies of many other DC application profiles.

In addition to specifying what set of terms is to be used in their metadata descriptions, the developers of a metadata application usually specify how their metadata descriptions are to be expressed for exchange between systems, i.e., the use of one or more formats for their metadata records. We have already noted that Dublin Core provides a number of binding specifications which describe how to encode DC metadata in a number of formats, and typically the application developer will select one of these bindings.

Two examples of widely used Dublin Core application profiles are the OAI-DC and RDN-DC application profiles, which we will now describe in more detail.

The OAI-DC Application Profile

The OAI-DC profile is the baseline metadata standard in the Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH) (Lagoze, Van de Sompel, Nelson and Warner, 2002). The OAI-PMH is a fairly simple protocol that supports the controlled transfer of metadata records over HTTP. The protocol allows the exchange of any metadata that can be serialized in an XML

format. The Dublin Core metadata standard has been widely implemented by services that make use of the, and the OAI-PMH specification requires that all OAI-PMH data providers must support the OAI DC application profile.

In this profile, a metadata description must consist of statements which reference only the fifteen properties of the Dublin Core Metadata Element Set. Properties are optional and repeatable, i.e., there is no requirement that all properties are referenced from statements in a metadata description, and the same property may be referenced in multiple statements. References to values must be made in the form of value strings, and neither vocabulary encoding schemes nor syntax encoding schemes may be used.

The RDN-DC Application Profile

The Resource Discovery Network (RDN) is a collaborative service provided for the UK Further and Higher Education communities which provides access to high quality Internet resources selected by subject specialists for their value in learning and teaching. The RDN makes use of OAI-PMH to transfer metadata records between partners, but rather than exchanging only OAI-DC records, the RDN deploys its own application profile, RDN-DC[34], which supports the creation of more expressive metadata descriptions tailored for the discovery requirements of the RDN (Day and Cliff, 2003). The profile references a subset of the properties provided by Dublin Core and requires the use of specific vocabulary encoding schemes for some of those properties; it also references some properties that were defined specifically for the requirements of the application.

Those local properties are defined and assigned URIs by the RDN in much the same way as the standard properties provided by the Dublin Core metadata standard and they are referenced in a metadata description, using a URI, in exactly the same way as a property provided by the standard. And indeed, although these properties were defined to meet the requirements of one particular community, they may be referenced by the developers of other DC application profiles developing applications for other communities if their usage is perceived as meeting some functional requirement.

The Singapore Framework for Dublin Core Application Profiles

As a result of intense discussions before and during the International Conference on Dublin Core and Metadata Applications in Singapore, September 2007, an overarching framework for Dublin Core Application Profiles was formulated and dubbed the *Singapore Framework* (Nilsson, Baker & Johnston, 2008b).

The motivation behind the development of the framework was to specify the necessary documentation needed for a Dublin Core application profile, and to ensure a certain level of homogeneity in the structure of application profile specifications.

The framework specifies five components in an application profile "documentation packet":

- **Functional requirements** specify the purpose of the application profile, and are used to understand the relevant uses of the application profile.

34 http://www.rdn.ac.uk/oai/rdn_dc/

- **A domain model** that defines the major entities and relationships described by metadata following the application profile. The entities in the model become the "described things" in the metadata records.
- **A Description Set Profile (DSP)** (as described in Paper 5) formally describes the metadata records that are valid instances of the application profile. A DSP describes what properties may be used, which vocabularies that are acceptable, and how a metadata record may be assembled according to the application profile. The DSP model is an XML-based constraint language, currently in working draft status at the DCMI.
- **Usage guidelines** describe more informally how the application profile is supposed to be used, and may include guidelines describing how to extract and interpret metadata from the described things. Usage guidelines are optional.
- **Encoding syntax guidelines**, important in some cases where the application profile is intended to be used in a particular syntactic context, describe any application profile-specific syntaxes or other guidelines for a particular syntax. Encoding syntax guidelines are optional.

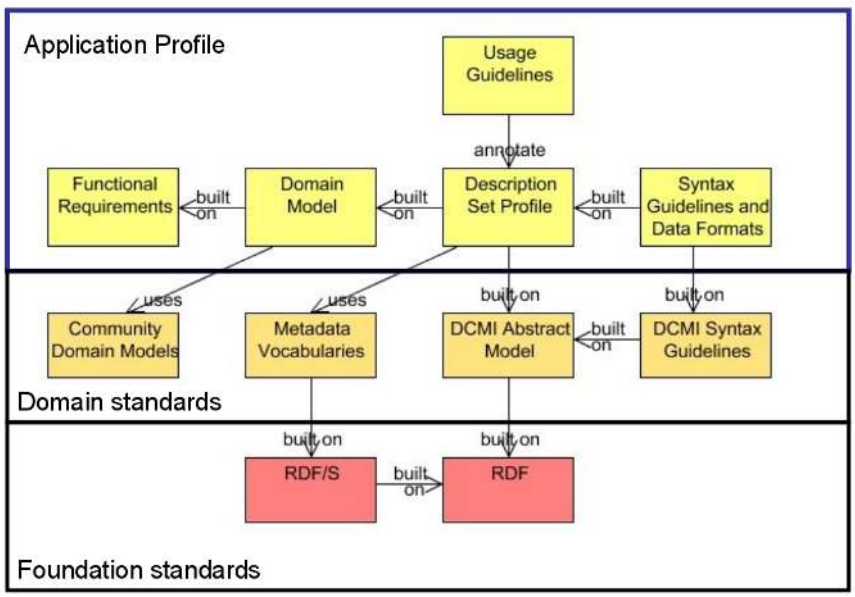


Figure 5.2: The components of the Singapore framework, and the underlying specifications

The relationship between the five component and the underlying specifications is described in Figure 5.2 (from Nilsson, Baker & Johnston, 2008).

Description Set Profiles are based on the metadata structure specified in the DCMI Abstract Model, which uses RDF and RDF Schema as a foundation.

Application profiles reuse one or more metadata vocabularies described in RDF Schema, defining classes and properties. They may also reuse widely recognized domain models (such as the Functional Requirements for Bibliographic Records (FRBR) being incorporated in many modern library metadata systems).

Description Set Profiles

The Dublin Core Description Set Profile model (Nilsson, 2008c and Paper 5) is designed to offer a simple constraint language for Dublin Core metadata, based on the DCMI Abstract Model and in line with the requirements for Dublin Core Application Profiles as set forth by the Singapore Framework. It constrains the resources that may be described by descriptions in the description set, the properties that may be used, and the ways a value may be referenced.

A DSP contains the formal syntactic constraints only, and will need to be combined with human-readable information, usage guidelines, version management, etc. in order to be used as an application profile, as described in the Singapore Framework. However, the design of the DSP information model is intended to facilitate the merging of DSP information and external information of the above kinds, for example by tools generating human-readable documentation for an application profile (see Paper 5).

A DSP describes the structure of a Description Set by using the notions of "templates" and "constraints".

A template describes the possible metadata structures in a conforming record. There are two levels of templates in a Description Set Profile:

- **Description templates**, that contains the statement templates that apply to a single kind of description as well as constraints on the described resource.
- **Statement templates**, that contains all the constraints on the property, value strings, vocabulary encoding schemes, etc. that apply to a single kind of statement.

While templates are used to express structures, constraints are used to limit those structures. Figure 5.3 (taken from Nilsson, 2008c) depicts the basic elements of the structure.

Thus, the DSP definition contains constructs for restricting

- **what properties** may be used in a statement and the multiplicity of such statements
- **what languages and syntax encoding schemes** may be used for literals and value strings, and if they may be used or not
- **what vocabulary encoding schemes and value URIs** that may be used, and if they may be used or not.

The DSP specification also contains a pseudo-algorithm that defines the semantics of the above constraints, i.e. how an application is supposed to process a DSP. The algorithm takes as input a description set and a DSP, and gives the answer *matching* or *non-matching*. In this way, a DSP defines the set of matching metadata records, making it usable for the kinds of metadata validation discussed in Hillmann &, Phipps (2007).

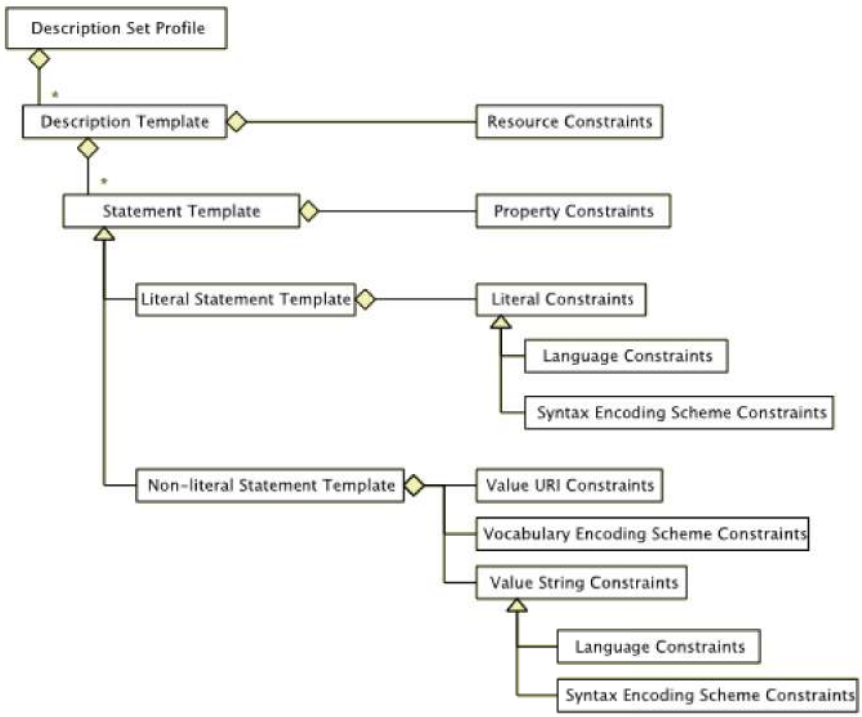


Figure 5.3: Templates and constraints in a DSP

The future development of Description Set Profiles is unclear. While there is a well-defined need for formal constraint languages on the level of abstract models, it's not clear that the current DSP approach is the most appropriate, and also not clear whether a constraint language is best applied to the DCAM, or directly to the underlying RDF model.

5.5.3 LOM Application Profiles

An examination of LOM application profiles reveals a slightly different approach. Instead of mixing and matching elements from multiple schemas and namespaces (Heery and Patel, 2000), it presents customization of a single standard to address the specific needs of "particular communities of implementers with common applications requirements" (Friesen, Mason and Ward, 2002).

That is, a LOM application profile is designed within the framework of the LOM abstract model. The terms referenced within a LOM application profile are terms of the type described by the LOM abstract model. A LOM application profile describes how the hierarchical structure described by the LOM standard is adapted to the requirements of an application – and indeed the nature of that adaptation is itself constrained by the LOM standard, which specifies data types and value spaces for each LOM data element and places some limits on the occurrences of LOM data elements within a LOM metadata description. This contrast between the scope of the LOM and Dublin Core metadata standards was noted earlier: while the Dublin Core standard specifies a set of terms to be used in metadata descriptions, it adopts a flexible approach to the ways in which those terms are deployed by an application. The LOM standard, on the other hand, both provides a set of data elements and defines a structural pattern of nested elements, with ordering and cardinality constraints, within which those data elements are deployed and interpreted. This set of standard structural constraints might be conceptualized as a “default” or “base” LOM application profile, one to which all other LOM application profiles must conform.

The most widely used mechanism for extending the LOM metadata standard is through the use of custom vocabularies to provide values for LOM data elements and the use of specified taxonomies within the LOM Classification element. Although the LOM abstract model does not require the use of globally unique identifiers for vocabularies and taxonomies, there are mechanisms provided (the “Source” sub-element within a Vocabulary data type item, and the “Source” element of the Classification category) which enable implementers to adopt conventions to distinguish between vocabularies, and to confirm that two references are indeed references to the same vocabulary.

Another common method of customizing LOM is through the tightening of structural constraints, such as making elements mandatory or to remove elements altogether, or putting an upper limit on the number of instances of a certain element. It is also common to produce additional guidelines for the usage of specific elements within the target community, something which is of particular interest for national customizations of LOM such as the UK LOM Core.

The LOM abstract model provides further possibilities for extensibility through the use of what it calls “extended data elements”, i.e. the use within a LOM metadata description of data elements other than those defined by the LOM standard itself. This combination of features – extensions and restrictions – presents unique challenges for the design of XML schemas for the LOM XML binding, as multiple notions of validation for the same application profile.

Three widely used LOM application profiles are the UK LOM Core, the RDN-LTSN LOM Application Profile and the Curriculum Online Metadata Schema, which we will now describe in more detail. The first two of these also demonstrate how one more generic application profile (the UK LOM Core) can form the basis for a second, more refined application profile (the RDN-LTSN LOM Application Profile).

UK LOM Core

The UK LOM Core LOM application profile is the result of efforts to promote common practice in the implementation of the LOM in UK educational contexts, in order to improve the ability of LOM metadata applications to exchange effectively the information required to support a number of basic functions[35] (UK LOM Core, 2005).

The UK LOM Core:

- specifies a "core" set of LOM data elements that should be present in LOM metadata instances
- provides information on the use and interpretation of LOM data elements within the UK context
- specifies a small set of vocabularies that should be used to provide values for some LOM data elements

RDN-LTSN LOM Application Profile

As noted above, the Resource Discovery Network (RDN) provides a Dublin Core application profile for metadata sharing between partners in the network. The RDN has also engaged in collaborative work with a similar network, the Learning and Teaching Support Network (LTSN) (since 2004 a part of the UK Higher Education Academy). Metadata sharing within this broader network was based on the use of a LOM application profile known as the RDN/LTSN LOM Application Profile (RLLOMAP)[36].

RLLOMAP is designed to support a specific set of functions to be delivered by the RDN-LTSN services. However, it is also designed to be compliant with the UK LOM Core. i.e., any LOM metadata description constructed according to RLLOMAP also complies to UK LOM Core. RLLOMAP specifies a set of LOM data elements and provides quite detailed guidelines for their use in the context of the RDN-LTSN community. It also mandates the use of some community-specific vocabularies (in addition to the LOM standard vocabularies) for some elements, and makes recommendations for the use of specified taxonomies for the LOM Classification element.

Curriculum Online Metadata Schema

The Curriculum Online service provides access to multimedia resources which support the curriculum taught in primary and secondary schools in England, and a metadata schema - an application profile of the LOM - was developed to support the specific requirements of this service. In particular, the schema supports the controlled classification of learning resources required to enable the rich searching and browsing functions that are provided to teachers and other users of the Curriculum Online web site (Department for Education and Skills, Simulacra and Schemeta, 2003a, 2003b).

Like RLLOMAP, the Curriculum Online Metadata Schema specifies which elements are required to occur in metadata descriptions and provides guidelines for providing values for those elements.

35 http://www.cetis.ac.uk/profiles/uklomcore

36 http://www.rdn.ac.uk/publications/rdn-ltsn/ap/

In addition, it defines some extensions to the LOM standard in the form of some additional data elements and vocabularies for the values of some of these elements. These “extended data elements” include a group of elements to support the description of the “Method of Delivery” of the resource, a group of elements that provide an indication of the cost of a resource, and an element to capture the name of the application used to create the metadata record.

5.5.4 Application Profiles in RDF

By contrast, there has been remarkably little work done in the context of RDF on application profiles, although the requirements on validation and coherence of RDF metadata has been steadily increasing. With the recent developments in the field of Linked Data (see Bizer et al, 2009), the concept has received increasing attention.

There are a few examples of application profile-like solutions for RDF. The Fresnel display vocabulary (Pietriga et al. 2006) provides a language for structured presentation of RDF triples. This fulfills only a small part of the functional requirements on application profiles, as the vocabulary is not expressive enough to allow validation of instance metadata or to provide the necessary support for creating and editing valid instances.

These aspects are managed in the SHAME metadata editor (Palmér et al., 2007), which was designed to provide capable methods for presenting and editing RDF metadata. The so-called “annotation profiles” used in SHAME correspond relatively closely to the Dublin Core Description Set Profiles, even though SHAME is based on an RDF query language instead of a custom constraint language.

Ratanajaipan et al. (2006) describes the potential of OWL as a language for describing application profiles. The method is interesting, but it should be made clear that OWL is used to describe semantics of RDF classes in properties in absolute terms, not in terms of domain-specific constraints. So, for example, two OWL-based application profiles for the same domain, using the same classes and properties but with, say, different cardinalities will result in a logical contradiction if for some reason the ontologies are loaded into the same system.

Van Assem (2010) (section 7.5) describes a solution to this issue based on application profile-specific subclasses, but this method runs into the issue that OWL semantics is based on an open world assumption. This leads to situations like the following: if a cardinality constraint is not met due to too few statements using the property, a processor is expected to infer the existence of an additional statement, not a cardinality violation. Similarly, if a cardinality constraint is not met due to too many statements with the same property, a reasoner is expected to infer that several of the values are, in fact, identical.

Thus, in order to use OWL as a validation tool, a completely alternative semantics needs to be superimposed on top of the OWL syntax[37]. This validation semantics would essentially create a parallel language to OWL, creating potentially serious interoperability problems when such ontologies are distributed.

For the above reasons, we can conclude that RDF is currently lacking an established format for defining application profiles.

37 As implemented, for example, by the Pellet Integrity Constraint Validator, http://clarkparsia.com/pellet/icv/

5.5.5 Application Profiles and Bindings

The developers of a metadata application – in most cases at least – also need to specify how metadata descriptions constructed according to the profile are to be expressed when they are exposed for exchange between systems, i.e. they need to specify the use of one or more formats for their metadata records. The developers will probably select one of the bindings specified by the metadata standard. In some cases they may develop a new binding to meet some particular requirements of their context (as is proposed by The International Press Telecommunications Council (2005)). Where application profile developers develop a new binding, they may choose to optimize that binding for the context of their application, e.g. by supporting only some subset of the constructs in the full abstract model of the standard. In any case, if a new binding is developed, it is essential that the developers make available a description of how the syntactic features they use are to be interpreted in terms of the standard's abstract model. They may choose to provide an algorithm or transformation by which a record conforming to their binding can be converted into a record using a standard binding. We will return to this important concept in section 6.4.

One promising framework for this kind of transformation specifically into RDF that is becoming increasingly popular is GRDDL, described in Hazaël-Massieux and Connolly (2005) as “a mechanism for Gleaning Resource Descriptions from Dialects of Languages; that is, for getting RDF data out of XML and XHTML documents using explicitly associated transformation algorithms, typically represented in XSLT”.

5.5.6 The Limitations of Mix and Match in DC and LOM Application Profiles

The first point that we have highlighted is that the DC and LOM concepts of the application profile are both rooted in the corresponding abstract models underpinning those 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 property is interpreted through the semantics of the DC 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 formalizing the way in which the implementers of metadata standards customize 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 models. For DC and LOM, the incompatibility of those abstract models means that the application profile construct is not sufficient to address the problem of how to use component parts of those two standards in combination.

5.5.7 Application Profiles in an XML context

XML-based metadata standards come with a natural method for designing application profiles and implement extensions, namely XML Schema or similar XML data description languages such as RelaxNG. For this reason, most, if not all, XML-based metadata languages rely heavily on XML schema for vertical harmonization.

The amount of syntactic interoperability and tool support achieved through reliance on the XML specification stack is significant – and this has been a powerful influence on application profile developments in other metadata standards. A good example is the Description Set Profile concept of Dublin Core, which has been designed to be convertible to XML schema when used together with an XML binding of Dublin Core[38].

However, as useful as XML-based application profiles are for XML-based metadata standards, they are still problematic to use as a basis for application profiles for standards based on an abstract model. Valid XML extensions or adaptations defined in an XML Schema might not be valid in the abstract model, and might therefore be unusable outside the XML context.

5.5.8 Summary of Application Profile issues

The following table, adapted from Nilsson (2010) summarizes the lessons from the above discussion:

38 An attempt at converting a DSP to the Schematron schema language can be found here: http://efoundations.typepad.com/efoundations/2009/09/experiments-with-dsp-and-schematron.html

Specification	*Application Profile support*	*Machine-readable 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 established notion of application profiles[39]	No formalism except OWL for ontologies.	Fully reusable.
ISO MLR	Profiles defined as hierarchically organized combinations of elements	No formalism.	Any part of an application profile can be reused separately.
RDA	Profiles defined in commercial RDA tool	Only in the commercial tool	Only within commercial tool.
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 (Description Definition Language).	Difficult to reuse extensions, though XML namespaces could help.

5.6 Summary

As we have seen in this section, vertical harmonization take many forms. We can, however, discern a few commonly discussed dimensions:

- An important kind of vertical harmonization concerns **application profiles**, where the harmonization focus is the interoperability of extensions and the validation of metadata records with respect to certain profile patterns. These needs are strongly present in several communities, with the RDF community conspicuously lacking much discussion on these concerns.
- Descriptions of **vocabularies** related to a particular metadata standard are also a common vertical harmonization issue. Examples include the IMS VDEX format for LOM vocabulary exchange, and RDF Schema for RDF vocabulary description.
- Another important vertical harmonization issue is simply interoperability of **metadata syntaxes** in the presence of an abstract syntax. We see these concerns in all metadata specifications where multiple syntaxes are present.

39 Somewhat similar functions can however be fulfilled by OWL ontologies, the Fresnel Display Vocabulary for RDF (Pietriga et al. 2006) or SHAME (Palmér et al, 2007).

- **Metadata semantics** on various levels of complexity and formalization are a central concern in those standards communities where machine-processable semantics is used, mainly Dublin Core and RDF. Examples include schemas (RDF Schema), ontologies (OWL) and rules (RIF), all of which describe metadata semantics on different levels of complexity and completeness. Another example is the focus within Dublin Core on documenting informal semantics of its terms for reuse of the DCMI Terms outside of RDF environments. (see for example the Dublin Core ISO standard, ISO 15836).
- Finally, **reuse of the metadata specification** in other specifications is a common pattern. In these cases, the metadata standard is used to support or complement the functions of a standard with a different purpose. Examples include the reuse of IMS metadata inside the IMS Content Packaging specification and the SPARQL RDF query language.

Now we can put these aspects aside and focus on cross-community harmonization.

6. Horizontal harmonization

Rather than vertical harmonization, this thesis focuses on harmonization that works between standards and across a variety of systems.

We will discuss three variants of horizontal harmonization:

- **Mappings or crosswalks** – harmonization through manually crafted metadata mappings between independent standards
- **Syntax-based combinations** – harmonization through manual mixing of metadata syntaxes
- **Vocabulary-based combinations** – harmonization through combinations and reuse of vocabularies across standards

6.1 Metadata Mappings/crosswalks

A different approach for improving metadata harmonization, often used for solving incompatibilities between metadata standards, is to produce *mappings* between the standards. This approach is broader than vertical harmonization in that it addresses the need to combine more than one specification or family of specifications. Many such systems have been implemented, with varying degrees of success (see for example Godby and Childress (2003)).

A mapping in this sense is defined as a translation that transforms metadata using one standard to metadata using another standard. Thus, these mappings are not based on reusing terms or mixing fragments, but on pure translation.

Mappings serve a useful purpose, as they address a pressing short-term need for translating between metadata formats. However, as a long-term solution to the harmonization problem, the approach suffers from a set of major problems:

- **Every mapping requires manual construction**, defeating the goal of machine-processability. In other words, mappings are a symptom of lacking semantic metadata interoperability. Such a mapping must also be actively maintained in order to continue to be useful, requiring even more work.
- **The differences in abstract models, terminology and vocabulary necessarily make mappings incomplete and sometimes ambiguous**, leading to imperfect interoperability. Mappings may be complex because they may have to operate not on stand-alone "elements" but on complex nested constructs.
- **Each new metadata standard requires a new set of mappings to each other relevant standard**, creating an astounding complexity. This can be somewhat relieved by mapping all standards to a common “base standard”. But as we have seen, the notion of a common base standard for metadata standards with incompatible abstract models is very problematic.
- **Semantic information tends to be lost.** Realizing mappings that are able to preserve not only the metadata constructs themselves but also their semantics (including refinements) is impossible in principle in many cases.
- **Mappings do not really solve the problem of *combining* parts from different standards**, only that of translating between standards.

Several of the difficulties are exemplified in Johnston (2005a) and Paper 2. The experiences from the LOM RDF binding in the latter paper shows that mapping between incompatible abstract models involves a complex re-modeling process, and that it may be difficult or impossible to make the resulting mapping bi-directional. These issues are also explored in Haslhofer & Klas (2010)

The conclusion, from the point of view of the questions posed in this thesis, is that while mappings can be used to solve immediate, practical, harmonization problems, they do not present a long-term, sustainable solution to the issue of lack of metadata harmonization.

6.2 Syntactical Combination

With the understanding of the role of abstract models reached in the previous sections, together with the description of the expression/interpretation process and the notion of interoperable processing, the problem of understanding what is really going on in the process of extending one metadata standard using terms from another metadata standard becomes much more evident.

6.2.1 Combining XML Languages

Let us recall Example 4.4 given earlier, in which a MODS XML metadata description was extended using a LOM XML fragment. We saw that assembling the combined metadata description seems to work, at least at a first glance.

The step of *interpreting* the format using the metadata semantics is the step that leads to difficulties when combining standards. The process is depicted in Figure 6.1. Application A produces MODS XML metadata, while Application C produces LOM metadata in the LOM XML format and inserts a fraction of that into the MODS XML metadata as in Example 4.4 above. Application B, which understands the MODS model, tries to interpret this combined XML document.

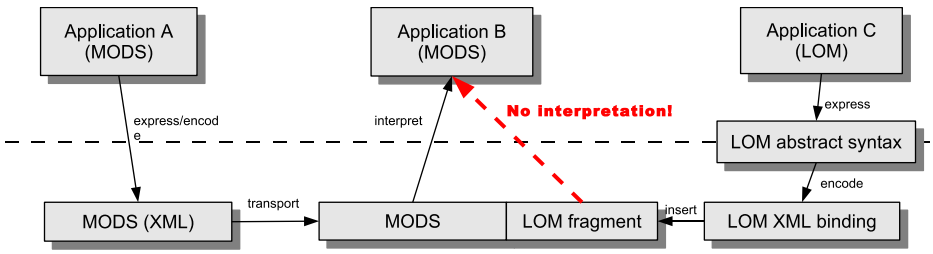


Figure 6.1: Combining the XML languages of LOM and MODS.

What now happens is that the LOM XML fragment is processed as pure XML, losing the information bound to the interpretation of the XML element in terms of the LOM abstract model. In this particular case:

1. The fact that lom:description is a LOM element, while lom:string represents a LangString value of that element is no longer available, as both are just ordinary XML elements.
2. The interpretation of multiple lom:strings as alternative localized versions of the same text is lost.
3. The interpretation of lom:description as the LOM element “Educational.Description” and not “General.Description” is lost, resulting in serious ambiguity in the interpretation of the XML element.

In short, a pure MODS processor will not interpret the metadata according to the intentions of the producer, creating a non-interoperable application. To solve this issue, the MODS processor needs to be extended to incorporate the semantics of elements from LOM.

There are two alternative approaches for such an extension:

1. Adding logic to process particular LOM XML elements in useful ways.
2. Adding a processor based on the LOM abstract model to process LOM extensions.

Both cases result in a new application, which will land in exactly the same interoperability problems when encountering a new metadata standard. This approach suffers from exactly the same issues as metadata mappings: each new standard requires new logic in all applications, combined processing based on fundamentally different models may result in data loss, etc.

Trying the other way around, extending LOM XML with MODS fragments, results in essentially the same kind of difficulties, only worse. The MODS XML fragment does not follow the LOM abstract syntax, and the semantics is therefore inaccessible to a LOM application. For example,

the MODS XML fragment uses XML attributes that are not part of the LOM abstract model and therefore cannot be meaningfully interpreted as LOM elements. The situation is summarized in Figure 6.2.

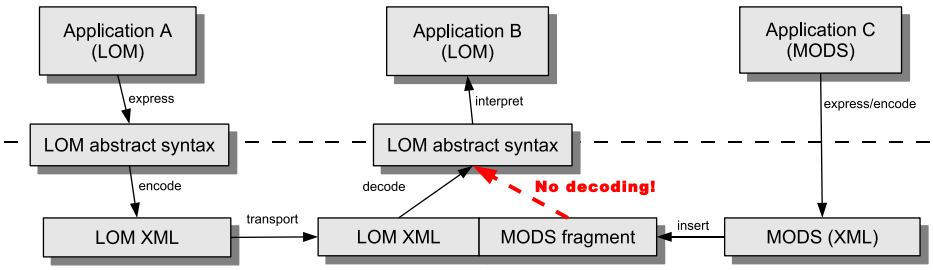


Figure 6.2: Extending LOM with a MODS fragment

To solve this issue, a LOM processor would need to go outside of the LOM abstract model and implement a MODS XML processor.

The result can be summarized in the following table:

Base format	***Extended with fragment from***	***Processable by LOM application***	***Processable by MODS application***
LOM XML	MODS	Only LOM part	No
MODS	LOM XML	No	Only MODS part

So it seems extending the current XML formats for LOM and MODS using terms from the other standard is a meaningless and purely syntactic exercise completely losing any semantics of the extension.

Combining LOM and MODS metadata leads to simple *data lacking semantics* rather than metadata, failing the harmonization test. This shows how metadata formats on their own are nonfunctional as a bass for improving metadata harmonization. Harmonization needs to take the metadata semantics into account.

The same is true for most XML languages for metadata – they are based on different, incompatible, and mostly non-overlapping semantics, and trying to combine them will not lead to improved metadata harmonization.

In many ways, the above exercises are similar to trying to combine, say, English and Chinese text in a single Unicode document and expecting the combination to make sense to a speaker of either language, or to combine source code fragments from two different programming languages based on the premise that they use the same character encoding. The only common thing is a low-level syntactic carrier, not capable of transmitting a combined understanding of the parts. The different metadata fragments might just as well be transmitted in separate XML files, and be consumed by two separate applications.

In order to achieve improved metadata harmonization, we must find a better approach.

6.2.2 Combining RDF Descriptions

The previous section described the results of combining two metadata standards with XML expressions. What happens if we try the same exercise with RDF versions of two standards, such as LOM using RDF and Dublin Core using RDF?

The first difference, as mentioned in section 4.2.4, is that the two cases of extending LOM with Dublin Core data or vice versa both lead to the same end result. There is only one resulting RDF description to consider.

The second difference is that being a metadata standard in its own right, RDF also brings us an abstract model with significant built-in base semantics. This means that RDF descriptions taken from different standards will be processable by a pure RDF application based on the RDF abstract model and semantics. The process is depicted in Figure 6.3.

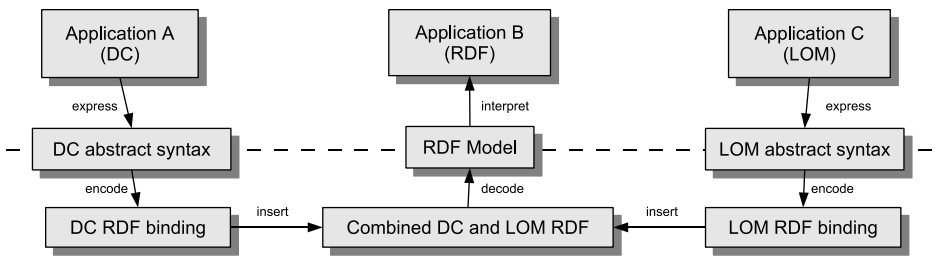


Figure 6.3: Combining RDF metadata from LOM and DC, interpreted through the RDF model.

Now, the Dublin Core abstract model is compatible with this base semantics of RDF. Any metadata conforming to the Dublin Core abstract model can be translated into RDF and back. As a consequence, Dublin Core applications (in the place of Application B in Figure 6.3) are actually able to process the LOM metadata expressed in RDF. LOM properties will be correctly understood as properties, and their values and datatypes will be processable. This means that any metadata standard that is completely independent of Dublin Core, but is still expressed in RDF, will be partially processable by a Dublin Core application. This is no coincidence – RDF and Dublin Core has been heavily influenced by each other during their development.

By comparison, a LOM application (in the place of Application B in Figure 6.3) will only be able to process those parts of the RDF files that have been mapped from LOM elements, and will not be able to understand, for example, Dublin Core metadata expressed in RDF.

The result can be summarized as:

Format	*Processable by LOM application*	*Processable by Dublin Core application*	*Processable by RDF application*
LOM+Dublin Core RDF	Only LOM part	Dublin Core part + LOM part	Dublin Core part + LOM part

6.2.3 LOM and RDF

As we have seen, translated LOM elements can be reused and processed by RDF and Dublin Core applications, but not the other way around.

The reason is that the LOM elements must be translated to RDF individually, in an idiosyncratic way – *there is no way to construct a general translation of the elements-in-elements-based abstract model of LOM into the property-value-based abstract model of RDF and back*. In other words, the abstract model of LOM and the base semantics of RDF are fundamentally incompatible (Nilsson, Palmér, Brase, 2003). This mapping therefore only understands LOM elements, and cannot specify how to interpret general RDF descriptions in terms of the LOM abstract model.

Conversely, the LOM RDF binding cannot specify how to translate extensions of LOM into RDF, as each of these extensions must be analyzed individually in order to determine how to represent them in RDF. The situation is depicted in Figure 6.4.

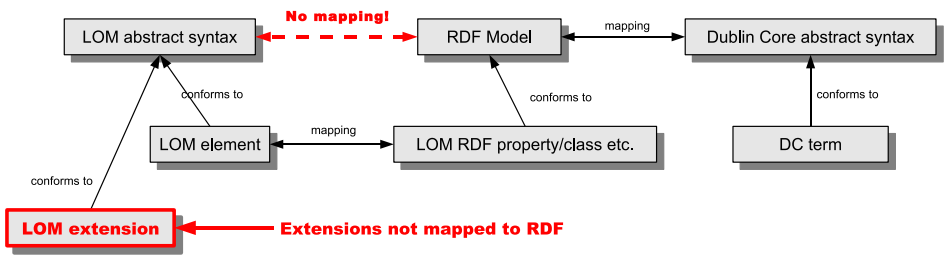


Figure 6.4: Issues when mapping LOM to RDF

What remains is a mapping on the individual element level – from a LOM element to a LOM RDF property and back. This is not to be confused with mappings between metadata standards as described in section 6.1, which deals with trying to translate between existing metadata specifications. Instead, this is a question of trying to represent LOM elements using a different abstract syntax, while retaining a compatible semantics.

The same incompatibility exists between any two metadata standards where one is based on an elements-in-elements model and the other is based on a property-value model, for example MODS and Dublin Core.

The above analysis shows why expressing LOM in RDF does not really constitute a “binding” in the sense that LOM in XML is a binding. RDF is more than a syntax, as it also carries semantics. A mapping from LOM to RDF is therefore not only a syntactic translation, but needs to be designed based on LOM RDF vocabulary and with metadata semantics of the resulting RDF

expression in mind. When that is done correctly, RDF applications can process LOM RDF instances without further adaption. Thus, we see that even with standards that use incompatible abstract models, using vocabularies as a basis for mapping is a feasible harmonization approach.

6.3 Reuse of Element and Value Vocabularies

As can be discerned from the above discussion, the notions of metadata "vocabularies" and metadata "elements" are somewhat ambiguous and are used differently in LOM and Dublin Core. However, as vocabularies are often a target for reuse across standards, they are highly relevant for horizontal harmonization.

In LOM, a "vocabulary" is a set of tokens with a specified "source" that can be used as values for certain elements. For example, the LOM element "Educational.Difficulty" can be used with values from a vocabulary specified in LOM, and containing the tokens "very low", "low", "medium", "high" and "very high". The "Source" must then be set to "LOMv1.0", to indicate that the values are from the LOM standard itself.

On the other hand, in Dublin Core a vocabulary can be one of two things:

1. A set of concepts as specified by a vocabulary encoding scheme. For example, the "dct:LCSH" vocabulary encoding scheme refers to the vocabulary formed by the set of Library of Congress subject headings. This corresponds closely to the notion of vocabulary in LOM, with the subtle but notable difference that Dublin Core deals with the concepts themselves (that may be referenced using a value string or a value URI, depending on the application), while LOM deals only with vocabulary tokens, i.e., opaque strings.
2. A set of metadata properties together with their definitions. For example, the Dublin Core Element Set, consisting of the 15 original Dublin Core elements, is such a vocabulary. The closest correspondence in LOM to this kind of vocabulary is the set of LOM elements.

In an attempt to generalize the vocabulary terminology, we will use the term *value vocabulary* to denote a designated set of terms used as values in metadata instances. The term *element vocabulary* (called "metadata schemas" in Haslhofer & Klas (2010)) will be used for the second kind – a set of terms used as building blocks in a metadata standard.

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.

As noted above, both LOM and Dublin Core have a notion of value vocabularies that include a notion of "vocabulary source". When specifying a value of a LOM element of the type "Vocabulary", the value may be accompanied with a "Source" string that gives an indication of the origin of the value, and therefore its interpretation. Similarly, Dublin Core uses the concept of vocabulary encoding schemes to specify the origin of a value, which may also be identified using a value URI. Being able to specify the source of a vocabulary is a requirement for interoperable metadata descriptions, and an important prerequisite for modular application profiles.

When it comes to element vocabularies, the situation is less clear. In Dublin Core, terms in element vocabularies, i.e., properties, must be assigned a URI to be usable in Dublin Core metadata descriptions. In this way, Dublin Core enables application profiles to mix Dublin Core properties with other properties in a controlled fashion, as the URI will allow applications to disambiguate between properties from different sources that are used in the same application profile.

However, the data elements defined by the LOM standard, as well as extended elements, are referenced not by globally unique identifiers, but by short human-readable labels like "Identifier" and "Context" (or "General.Identifier" and "Educational.Context", if their category is taken into account). There is an implicit assumption that a human reader or an application reading or processing a LOM metadata description will be able to determine from some contextual information that the data element is that data element defined by the LOM standard.

Perhaps for this reason the term "LOM application profile" appears to have been applied principally, though not exclusively, to those descriptions of LOM implementation that are limited to the data elements specified by the LOM standard, with extensibility restricted to the specification of value vocabularies and taxonomies. Where extended data elements are used in LOM application profiles, the implementer assigns labels to distinguish their data element names from those used for data elements defined by the LOM standard and in other LOM application profiles – but since these are simply arbitrarily chosen labels, rather than identifiers assigned with an identifier scheme, they can not be guaranteed to be unique. For this reason, LOM lacks support for machine-processable reuse of element vocabularies across application profiles.

The situation is aggravated by the fact that the LOM XML binding *does* provide namespace URIs for both the LOM elements and for elements used in extensions to LOM. *But as these URIs are not part of the LOM abstract model, they cannot be used outside the LOM XML binding to refer to the relevant LOM element.*

6.3.1 Reusing “Elements” Across Metadata Standards

What we have seen in this chapter is that mixing different metadata standards in 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 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 are created within different abstract models and uses 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": and it does not mean that it is 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 the analysis in this section, is that we must not confuse the components used in a metadata syntax and 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:

- **Mixing fragments of two metadata formats created to conform to different abstract models**, such as MODS 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 nonsense metadata constructs (Johnston, 2005a).
- In general, **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 we have seen, this leads to nonsensical metadata constructs, as the URIs of Dublin Core and of LOM must be interpreted in terms of different abstract models.
- **Mixing two different bindings of the *same* standard**, when those two bindings apply *different* interpretations to the use of *similar* components in the metadata format. This is the case with the LOM XML binding, which must be interpreted using a different set of rules than the RDF/XML serialization of the LOM RDF expression, though they contain component parts that may be confusingly similar.

So 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.

In conclusion, we see that in order to reuse components of different standards in a machine-processable way, 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.

The analysis of value and element vocabularies shows how the most important carriers of metadata semantics are element vocabularies, alongside the abstract models. The abstract models are methods of combining the individual semantics of metadata elements to form meaningful complete metadata descriptions. At the same time, the semantics of elements require a context in the

form of a metadata standard to be defined – element semantics cannot be defined in isolation. Rather, metadata elements are similar to words in a human language carriers of meaning, but dependent on a language to be interpreted correctly.

6.3.2 Summary of Element Vocabulary Features

Adapted from Nilsson (2010):

Specification	*Method for defining element vocabularies*	*Element identification*	*Element relationships*
IEEE LOM	Defines element vocabularies by describing the element placement in the metadata hierarchy.	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.
ISO MLR	Defined using ISO MLR-specific “data element definition” loosely based on RDF	ISO identifier	Allows refinement using “sub property” relation
RDA	No formal method defined	No formal identifier	“Sub-elements” corresponding to tree-based substructures, and “element sub-types” corresponding to sub-properties.
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 (Description Definition Language).	XML name	Allows syntactic refinement through subclassing in DDL, as well as sub-structures.

6.3.3 Summary of Value Vocabulary Features

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, Source/Value pairs, string tokens and natural language strings.

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.	Refers 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.	Refers to values primarily using URIs.
ISO MLR	Defined using ISO MLR-specific “data value definition”	Refers to values using ISO identifiers
RDA	Has no formal way of defining vocabularies	Refers to values only using textual label
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.

6.3.4 Summary of Element Identification Features

Different methods of identification imply different levels of precision, support for multilingualism and application independence. In order of decreasing precision:

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.

6.4 Semantic Embedding

As we have seen in the sections above, successfully combining metadata descriptions relies on a meaningful interpretation of individual metadata elements, as well as of metadata structures. A purely syntactical mixing approach is not sustainable.

At the same time, we have seen that metadata standards with differing abstract models may still be fully compatible (such as RDF and Dublin Core). In summary, there are two necessary components for combining metadata conforming to one metadata standard with another:

1. A syntactical mapping translating the structure from one metadata standard to the other. Simply mixing independent syntaxes is meaningless - there needs to be a mapping between the syntaxes of the two standards so that the result makes sense to the receiving application. As between Dublin Core and RDF, this mapping is preferably on the structural level, rather than on an element-by-element basis, since the latter makes both complete mappings and two-way mappings very difficult. However, the LOM example shows that more idiomatic element-based translations also make sense.

2. Semantic coherence. Regardless of whether you interpret the metadata before or after the mapping, the interpretation needs to stay the same. The mapping must be designed to preserve the semantics of the original metadata, not only transfer an opaque structure.

This summarizes the issues we have encountered when mixing syntaxes without consideration of the semantics. We will use the term “semantic embedding” to denote the combination of metadata from two standards into one of the standards with consideration of the semantics.

This notion is closely modeled on the notion of embedding for comparing programming languages defined in Shapiro (1989) and refined in Shapiro (1991). The application here is different, since we will not use embedding for comparison, but as a harmonization tool. The metadata languages we analyze are not Turing-complete programming languages, meaning that the notion of *basic embedding* as defined by Shapiro is useful for our purpose.

6.4.1 Semantic Embeddings and Semantic Embeddability

We can present this conclusion using standard mathematical notation of a *commutative diagram* as in Figure 6.5. Let $map_{A,B}$ be a mapping from the set of metadata fragments conforming to metadata standard A to the set of metadata fragments conforming to metadata standard B. Let $I_A(m)$ be the interpretation of a metadata fragment $m \in A$, and $I_B(n)$ be the interpretation of a metadata fragment $n \in B$. Let $\nu_{A,B}$ be a natural[40] translation of an interpretation of a metadata fragment from standard A to an interpretation of a metadata fragment from standard B.

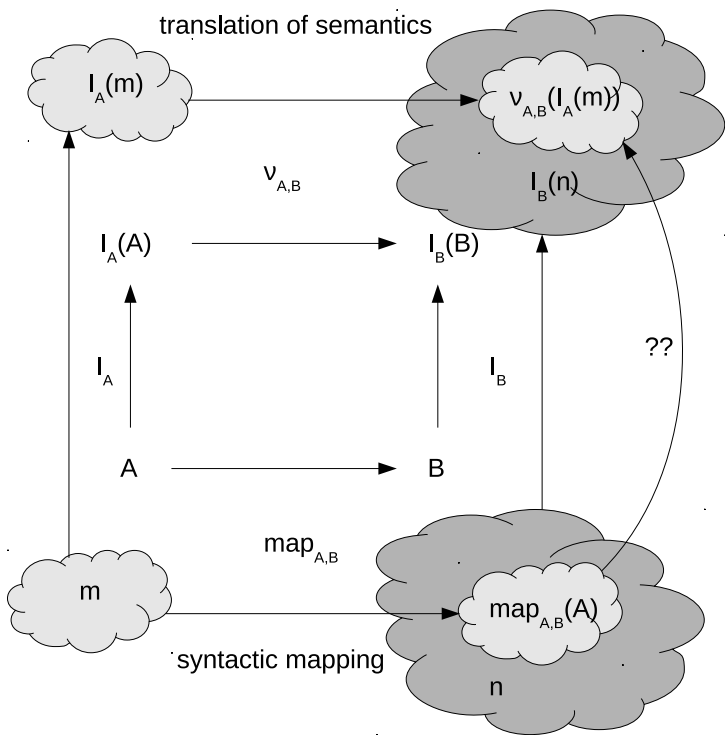


Figure 6.5: When the diagram commutes, A is semantically combinable with B

40 We use “natural” to mean that a human would understand the translated interpretation to mean the same as the original.

Then we can say A is *semantically embeddable* into B using $map_{A,B}$ and $\nu_{A,B}$ if

$$I_B(map_{A,B}(m)) = \nu_{A,B}(I_A(m)) \text{ for all metadata fragments } m \text{ conforming to } A$$

In plain English, A is semantically embeddable into B if the interpretation of a metadata fragment from A is the same, regardless of whether we interpret it directly or if we first map it to B. The important notion in these embeddings is therefore that the embedding preserves the metadata semantics. If semantics is lost or distorted, the semantic embedding fails.

We say that such a semantics-preserving pair of mappings, $map_{A,B}$ and $\nu_{A,B}$ is a *semantic embedding*[41].

Examples 4.3 and 4.4, where a metadata fragment from LOM is inserted into MODS and vice versa, are examples of embeddings that are not semantic embeddings, as the fragments lack interpretation when inserted into the other standard. On the other hand, the embedding of a LOM fragment into RDF in Figure 4.2 is an example of a semantic embedding of LOM metadata into RDF, as the interpretation of the LOM statements in RDF will be identical to the interpretation of the original LOM fragment (expressed in LOM XML).

We use the term *informally semantically embeddable* if the standards are semantically embeddable using informal semantics I_A and I_B, and an informally defined translation $\nu_{A,B,}$ while we use the term *formally semantically embeddable* if the standards are semantically embeddable using an formal semantics and a formal translation. If A is semantically embeddable into B, and B into A, then we say that A and B are *mutually semantically embeddable*.

Examples include LOM, which is informally semantically embeddable into RDF using the LOM RDF mapping. There is a subset of RDF, consisting of the image of $map_{LOM,RDF}$ which is informally semantically embeddable into LOM. Dublin Core and RDF, on the other hand, are mutually formally semantically embeddable in their entireties using the standard embedding.

Semantic embedding of one standard into another is a faithful metadata combination of the form that metadata harmonization requires, even though it is built on a mapping rather than direct syntactic combination. All essential information contained in the metadata is preserved, unlike the case with the metadata mappings discussed in section 6.1 or the syntactic metadata combinations in examples 4.3 and 4.4.

6.4.2 Semantic Embeddings and Harmonization

In order to avoid a situation where a combination of metadata standards is treated simply as multiple independent parts, it is essential that the processing of combined metadata is made within a single semantic framework.

An example of an undesirable situation is the repository system Fedora, which claims that "Metadata [...] in any format can be managed and maintained"[42] - which on the surface seems to address harmonization. But in practice, Fedora stores metadata from different formats in separate containers, and there is no semantic combination of the metadata parts produced, even though the separate metadata containers might describe the same things.

41 Shapiro (1989) uses the term "basic embedding" for this construct.

42 See http://fedora-commons.org/about/features

An example of a metadata system performing a similar function as Fedora, but operating in a harmonized way is the SCAM RDF-based repository system, described in Palmér et al (2004) and demonstrated in a federated setting in Manouselis et al. (2008). In this repository, federated metadata in a variety of formats is semantically embedded into the RDF repository, making all of the metadata available for processing.

Using the definition of metadata harmonization, we can therefore say that the metadata standards S_1 to S_n are *harmonized* if there is a metadata standard S such that all of S_1 to S_n are semantically embeddable into S using some mapping. That is, we reach harmonization if all standards under consideration can be embedded into a single standard while preserving the semantics of the metadata.

We can now express the optimal level of metadata harmonization of S_1 to S_n as the level when two systems both process metadata using such a standard S. To increase harmonization, our goal now becomes to find the patterns that metadata standards should follow that increase the probability of creating a faithful semantic embedding into S.

6.4.3 Automatic Semantic Embeddings

In the context of a multitude of standards, it is desirable that semantic embeddings can be triggered automatically by software. There are two aspects of this automation:

1. The embedding $\mathsf{map}_{A,B}$ from standard A to standard B. In many cases, this needs to be a manually triggered process. There are important exceptions, however, such as GRDDL (Hazaël-Massieux & Connolly, 2005) a W3C recommendation for automatic triggering of conversions of XML languages to RDF.
2. The interpretation of new and unknown vocabulary when the syntactic mapping has been performed, in particular the semantics of element vocabularies. Without machine-processable element semantics, interpretation of new elements must be done completely manually on a case by case basis.

The latter, interpretation of metadata elements, is the key issue in semantic metadata interoperability for standards based on an abstract model. To exchange semantics, two systems only need to exchange element semantics, as the abstract model will be known. Thus, we again see the fundamental role that semantic metadata interoperability using machine-processable element vocabularies plays in harmonization.

6.5 Addressing the Harmonization Issues

The above analysis shows that there are many difficulties on the road towards improved metadata harmonization. In Nilsson (2010), five main areas of harmonization are identified: identification harmonization, abstract model harmonization, vocabulary harmonization, application profile harmonization and syntax harmonization. We present an summary of these findings here, adjusted with the additional findings presented in this thesis. An important common thread in these findings is the focus on automation of metadata semantics.

6.5.1 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 using locally scoped tokens work locally and in well-defined communities, but on a global scale, global identification is necessary. At the same time, locally unique tokens and natural language strings play an important role in interacting with people and other systems.

A related issue is when element identification depends on the placement of an element in a hierarchy, as in the LOM standard. Element vocabularies need to be reusable outside their original context to be useful targets for harmonization.

Approach

- Encourage the specification of URIs for values in controlled vocabularies.
- Provide mappings from such URIs to relevant tokens and natural language strings.
- Encourage the specification of URIs for metadata elements.
- Use web best practice to provide machine readable documentation of the vocabularies based on their URIs (as documented in Sauermann & Cyganiak, 2008)
- Make sure elements are syntactically context-independent, enabling them to be used in new contexts and combinations.
- Make sure elements are semantically context-independent, ensuring that they carry their own well-defined semantics.

6.5.2 Abstract Model and Syntax

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.

We have seen how the most important aspect is semantic embeddability, rather than using exactly the same abstract model. Still, embeddability does not come for free, but places strict demands on metadata standards.

Approach

- Encourage standards to base themselves on an abstract model. Basing mappings on abstract syntaxes makes mappings cleaner, easier to verify, more robust and less tool-dependent.
- Focus on reuse of elements rather than translation of instance data. As described sin section 6.1, except for highly similar standards, this tends to lead to incomplete and only partly semantics-conserving mappings. Instead focus on creating a mapping that reuses, in a semantically coherent way, the elements of one standard in the other.

- Discourage the introduction of fundamentally new abstract models into the domain, as this further fragments the community and increases the difficulty in creating combinable standards. Of particular worry in the domains studied in this thesis is the work on ISO MLR.
- Make sure that concrete metadata syntaxes are firmly grounded in an abstract model, and that, conversely, the abstract model is considered before the syntax when developing metadata specifications. A metadata syntax is useless without a processing model, and such a model must be based on the abstract model of the metadata standard.
- Ensure the abstract models are semantically embeddable into major metadata frameworks in use, such as RDF. Encourage the specification of such embeddings as part of the standard.

6.5.3 Vocabulary Models

There is no strong requirement for a single value vocabulary model, since the major harmonization issue relating to value vocabularies is value *identification*.

Values tend to be atomic concepts, not depending on the context in which they appear in instance metadata, and varying models therefore do not contribute in a crucial way to harmonization issues. While metadata "elements" (LOM-like or RDF-like) depend heavily on the metadata model, the meaning of a value is self-contained, or at least contained in the definition of the vocabulary.

Therefore, using RDF Schema, SKOS, IMS VDEX or similar techniques all work for basic value vocabulary description. However, without the vocabulary semantics being available for interpretation, much of the semantics will be out of reach for machine-processing, decreasing metadata harmonization. It is therefore important that vocabulary models be semantically embeddable into major metadata standards.

For element vocabularies, the case for a common model is significantly stronger. The issue is tightly linked to the embeddability of metadata standards, as element vocabularies are very important carriers of metadata semantics.

In the analyzed specifications, essentially two element vocabulary models are used: RDF Schema and XML Schema. 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.

In addition, machine-processable formats for element vocabularies are a prerequisite for enabling automatic processing of composite metadata, and in particular semantic metadata interoperability.

Support for ontologies require formally specified element vocabulary semantics. In all, this points towards a need for formal element vocabulary semantics expressed in machine-processable form.

Approach

- Ensure element vocabularies adhere to an abstract model so that the rules for reusing them are clear.

- Prefer machine-processable expressions of element vocabularies, with explicit semantics.
- Use established formats for expressing and publishing value vocabularies.
- Prefer formats for element and value vocabularies that are semantically embeddable into major metadata standards.
- Make sure that value vocabularies are defined without strong dependence on abstract models.

6.5.4 Application Profile Models

Application profiles that work across standards 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.

Application profiles are essentially syntactic constructs, combining metadata from different standards but letting the semantics be handled by the underlying metadata standards. Therefore, it is essential that application profiles are firmly based on the relevant syntax.

Approach

- Use models for application profiles that are independent of particular element vocabularies.
- Base application profiles on abstract syntaxes rather than concrete syntaxes when possible, to make sure the profile is usable across concrete syntaxes while still staying within the limits of the metadata standard.

With these conclusions in mind, we are now in a position to formulate the necessary components of a framework for metadata harmonization.

7. Towards a Harmonization Framework for Metadata Standards

If we try to look forward into the future of metadata standards in the web environment, it seems clear that an improved approach to metadata standardization is needed in order to fulfill the metadata harmonization requirements we set forth early in this thesis.

There have been initiatives to develop a common abstract model that covers both LOM and Dublin Core models without any form of mapping, but unfortunately it seems to be impossible to arrive at such a model without re-engineering at least one standard to retrofit it to the new abstract model, which naturally is a major undertaking. Similar conclusions can be drawn regarding other combinations of standards – in general, their models are not directly usable in combination.

In order to achieve a better level of harmonization between metadata standards we instead need to focus on their semantic embeddability. This way, information expressed using one standard will be available to applications using any standard it can be embedded into.

In a situation with a number of metadata standards that are targets for harmonization, the only feasible approach for a system that wants to implement them interoperably is to find a single standard into which all of the others can be embedded while retaining semantics.

A more realistic long-term goal is therefore to use the notion of semantic embeddability and make sure we can map the different standards to such a base standard. Based on the analysis in the previous sections, we can conclude that a harmonization framework built on the foundation of a single solid abstract model is a desirable goal for future metadata standards.

In order to provide support for automatic semantic embeddings from a variety of metadata sources, its important that such an abstract model comes with support for machine-processable semantics for the metadata elements. A *formal* semantics would, in addition, enable building ontologies based on the abstract model.

The metadata standards in current use go some way towards the fulfillment of this goal, but they operate mostly in isolation from each other, and one important component is missing: a metadata harmonization framework that presents the proper context for metadata standards. Throughout this thesis, we have gathered enough requirements to be able to put together a vision of such a framework, building on the work presented in Paper 4.

The main purpose of this section is to create a model that better reflects best practice when it comes to formulating metadata standards. The current situation, where metadata standards try to standardize very different things (model, syntax, semantics, vocabularies, etc) is an important source of harmonization troubles. With the knowledge we now have, we are in a position to give guidance to metadata specification developers about what kind of specification they should be developing.

One important part of this work is to improve the vocabulary we use when discussing metadata specifications. For example, the current widespread use of the terms “metadata standard” or “metadata schema” needs refinement.

7.1 Basic Structure of the Metadata Framework

The most central distinction in the proposed framework is based on an analysis of the respective roles of the specifications involved in the creation of metadata. There are three different categories in the model:

1. **The core abstract model.** We concluded in section 6.4.2 that harmonization requires a single abstract model that can be used as a mapping target for other standards. This model encompasses an abstract syntax, a model for element vocabulary definitions, and the corresponding semantics. As we have seen, this core is the basis for harmonization, and each such incompatible core will create an incompatible metadata island with respect to harmonization.
2. **Technical metadata specifications related to the core model.** These specification can be defined independently, and harmonization does not suffer if there are several specifications filling the same function in the metadata universe. This category includes specifications such as metadata syntaxes and application profile models.
3. **Domain-specific definitions.** On this level we find specifications that define specific metadata resources, such as vocabularies or application profiles. These definition generally just apply a specification to produce a set of conforming entities.

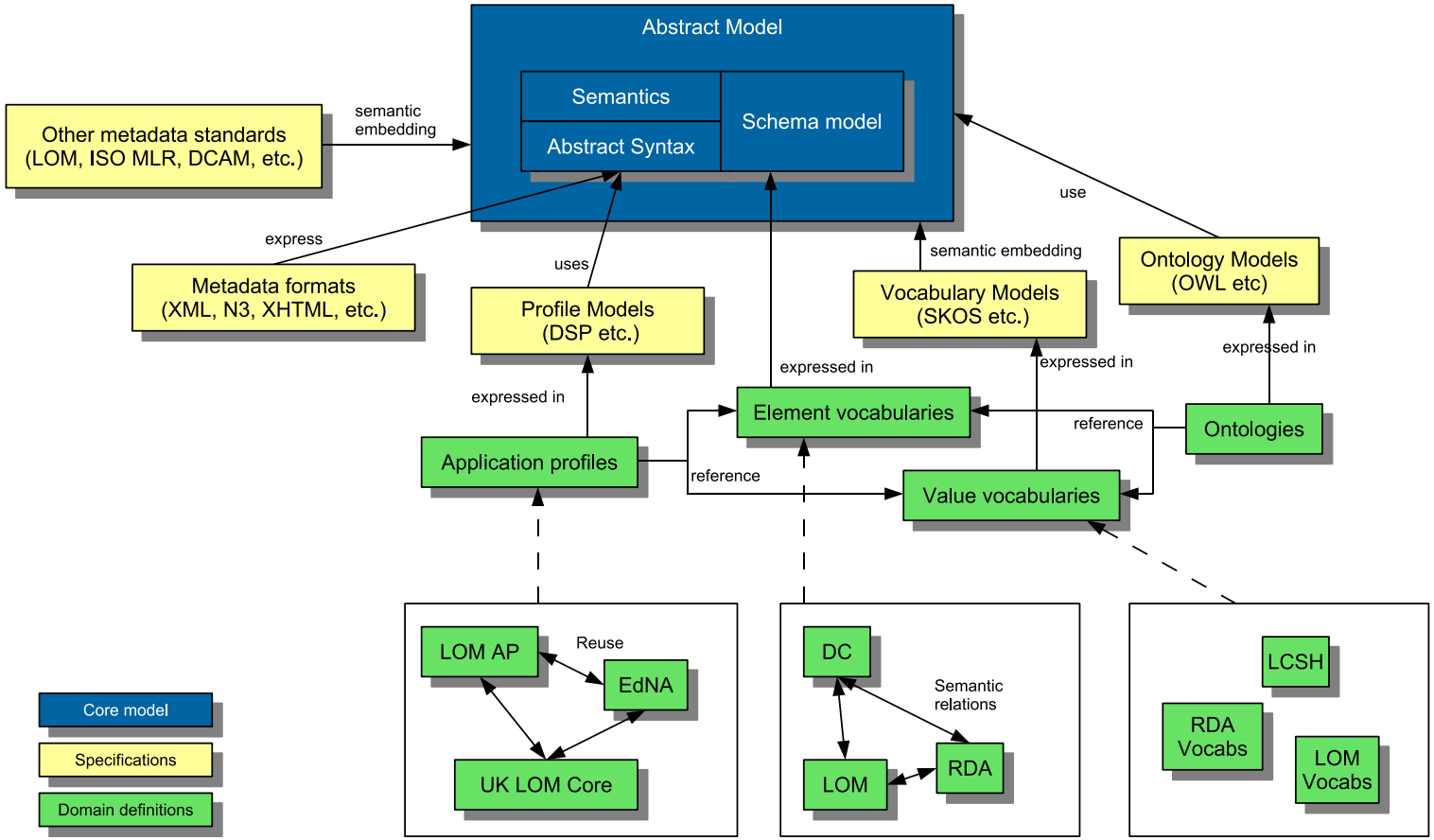


Figure 7.1: A possible structure of a future metadata standardization framework.

7.2 The Core Model

At the core of the proposed harmonization framework is a set of specifications that together provide the necessary scaffolding for harmonizing metadata standards: a common abstract model based on an abstract syntax, and a schema model for describing element vocabularies.

7.2.1 A Common Abstract Model

The basis of the envisioned harmonization framework is the abstract model. As we have seen, the incompatibilities of abstract models are the most significant stumbling blocks for metadata harmonization. The development of a common abstract model for metadata is therefore of central importance if we are ever going to experience true metadata harmonization.

Agreeing on such an abstract model is a major undertaking, not so much because of the technical difficulties, but because of the lack of coordination between the major standardization organizations involved. Still, the process is necessary and will give a number of tangible benefits, including:

- A single set of format bindings. Contrast this with the current situation, which requires every metadata standard to have its own set of format bindings. This will make life easier not only for metadata standardization bodies, but also for applications that will only need to support one format.
- A single framework for extending and combining metadata from different standards. This will enable standardized principles for the construction of interoperable application profiles.
- A single storage and query model for very different types of data and schemas. For example, storing metadata from different specifications in the same database is straightforward. Implementing searching that includes dependencies between metadata expressed in different schemas is simplified.

Thus, the development of a common abstract model leads the way towards support for all our metadata interoperability principles described in section 2.4: extensibility, modularity, refinement, multilingualism and machine-processability.

7.2.2 Schema Model

As discussed earlier, an abstract model relies on an abstract syntax with well-defined semantics. An important lesson from the discussions about semantic embeddings was that the core model also must include a model for specifying element vocabularies. We will call such a model, allowing the definition of metadata element and value vocabularies that fill the abstract model with metadata terms and relationships between terms, a *schema model* (the term “schema definition language” is used in Haslhofer & Klas (2010))

In a metadata harmonization framework supported by a common abstract model, the work of defining new metadata terms is much reduced. As the “grammatical structure” of metadata descriptions is already laid down, the only thing needed is to fill the abstract model with specific

terms. In order to do so, we need a language for describing metadata vocabularies. RDF Schema is one such schema language, and the only widely used existing language that matches the requirements presented here.

The main benefits of developing vocabularies in a common framework are:

- Clear guidelines on how to create and maintain customized metadata element vocabularies. There is currently some confusion on how to best produce element vocabularies, much due to the differing fundamental principles for vocabularies in the different metadata standards.
- Fine-grained control over relationships between terms from different standards, including refinement and partial mappings. Automation of interoperable metadata management will be greatly improved, and metadata vocabularies will be able to build upon each other.
- Reuse across standards will be much simplified. As an example, many elements in the LOM standard are not specific to learning, and have similar counterparts in other standards. In a common framework, the LOM elements will be made into a fully-fledged element vocabulary capable of being extended, refined and semantically annotated. The semantic relationships to terms in these other standards can be made explicit and machine-processable.

One interesting consequence of a common element vocabulary framework is the possibility of unexpected collaboration. That is, as others specify relationships to a vocabulary, new relations between resources will start to appear, and applications will be able to process metadata elements that had no previously declared semantic relationships.

7.3 Metadata Specifications

This category contains the important technical specifications that are not part of the core model. Because they are not part of the core model, there may be overlapping specifications or specifications filling the same purpose in this category.

These specifications are generally relatively stable technical specifications designed to create tool support for working with metadata. There are four major kinds of specifications in this model: metadata syntaxes, application profile models, ontology models, and semantic embeddings of other metadata standards.

7.3.1 Metadata Formats

These include bindings of the abstract syntax to a set of formats and systems, including XML, database layouts and programming languages. We will not dwell on the relationship between syntaxes and the abstract model, since this has been thoroughly addressed in section 4.

7.3.2 Profile Models

Application profiles specify usages of metadata vocabularies in complex combinations. As we have noted, the LOM standard contains a basic application profile, and this aspect of LOM needs to be separated from the definition of the element vocabulary consisting of the LOM elements.

Frameworks for expressing application profiles will be necessary building blocks for the construction of reusable application profiles. We envision several such frameworks, some tied to a specific metadata format, some operating at the level of the abstract model, so that the application profile can be reused in all metadata formats.

An example of a syntax-specific tool for building application profiles is XML schema, which has even been used with varying degrees of success to specify RDF-based application profiles such as Simple Dublin Core in RDF/XML[43].

Promising work on machine-processable application profiles can be seen in the DSP model (discussed in section 5.5.2) and "Guidelines" (2005). There are also other initiatives for such frameworks, but none are yet in widespread use.

There is no harmonization danger in letting multiple application profile models co-exist. Such models are usually tied to a specific tool set, a particular set of technologies or functional requirements or a specific community (such as the Dublin Core-based Singapore Framework), but at the same time not involved in the basic mechanisms of semantic metadata combination.

7.3.3 Vocabulary Models

Vocabulary models are used to describe metadata value vocabularies in order to increase interoperability between systems using the vocabulary. We concluded in the discussions in section 6.5.3 that a common vocabulary model is not a central requirement for metadata harmonization. Instead, a plethora of vocabulary description methods can coexist without any significant harmonization issues.

Value vocabularies can generally be used in the context of other metadata standards without any need for specially crafted vocabulary description formats. An example is LCSH, the Library of Congress Subject Headings, that have been used in Dublin Core metadata without issues for many years.

Still, such vocabulary must have a basic form of interoperability with the Schema Model. The core requirement is a compatible method for identification, so that vocabularies can be used in metadata instances without unnecessary ambiguity.

A second requirement is that the vocabulary descriptions are semantically embeddable into the core model. In this way, vocabulary descriptions are viewed as just another form of metadata, describing vocabulary terms. We want the semantics of this metadata to be available for processing.

We thus require the same level of harmonization from vocabularies as we do from other metadata standards, with the additional requirement of interoperable identification.

43 http://dublincore.org/documents/2002/07/31/dcmes-xml/#appB

7.3.4 Ontology Models

Another kind of specification are ontology models, such as OWL. While such models require a formal semantic model in the core model in order to be mathematically solid, the existence of multiple ontology models is not a harmonization issue. Indeed, OWL itself exists in several variants such as OWL-DL and OWL-Full with very different characteristics and application areas.

7.3.5 Semantic Embeddings of Other Standards

In order to use this framework together with standards that are not implemented using the framework, we require semantic embeddings to the abstract model. Because much of the semantics of the resulting metadata is carried by the target metadata elements, such mappings will need to use element vocabularies that conform to the schema model of the core model, at the same time as they must capture the full semantics of the original metadata.

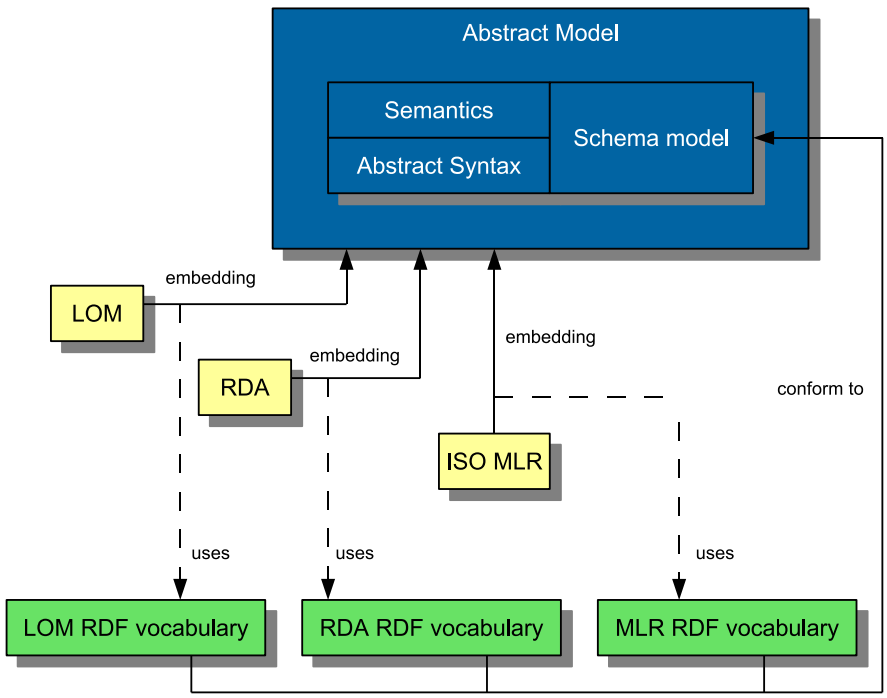


Figure 7.2: Combining standards using element vocabularies

As is depicted in Figure 7.2, this means that semantic embeddings require each source metadata standard to provide a corresponding element vocabulary. This precise situation has been realized in at least two cases:

1. LOM RDF binding – where the mapping was split in two parts: a formal LOM RDF vocabulary standard, and a separate recommended practice using this vocabulary to map from the full LOM model to the Dublin Core Abstract Model (and, implicitly, RDF).
2. RDA, where an RDA vocabulary expressed in RDF has been developed, as a prerequisite to future mappings from RDA to RDF.

The same pattern is expected for other semantic embeddings, such as from ISO MLR to RDF.

7.4 Domain-specific Definitions

In the third category of specifications and standards, we have the vocabularies, application profiles and ontologies produced by various projects and communities.

These are, if implemented using this framework, essentially applications of the technical metadata specifications, and should be relatively easy to produce for a knowledgeable individual.

More importantly, these specifications should not introduce new technologies or require reprogramming of application software, but should be “consumables” from the point of view of a metadata implementor. Pushing down metadata development to the “user” level in this way, removing the burden on vocabulary developers to define their own technical specification, is a major achievement if successful. We can already see this happening in the context of the Semantic Web.

7.5 Which Core Model?

We have seen clear evidence that the RDF family of specifications provides an abstract framework of the kind envisioned here, including a formal semantic model, a vocabulary description framework (RDF Schema) and well-designed integration with web technologies. However, it remains to be seen if using RDF will be acceptable as a foundation for the wide set of applications that use LOM, Dublin Core, RDA and others. RDF was designed for the open world of the Web, assuming an unreliable, distributed system with a multitude of sources (Fielding & Taylor, 2000), while the metadata that are of concern to us are important for a wide range of systems that are not restricted to web-oriented systems.

Other candidates for a core model do exist. ISO Topic Maps (ISO/IEC 13250) is a metadata specification with a relatively wide user base, but has not seen the same level of tool support or usage as RDF. It also lacks a formal semantic model, making it difficult to use as a basis for ontologies. Other descriptive standards such as KIF are strong on ontology, but weak when it comes to integration with web technologies and the open world assumption.

Dublin Core has proven that simple metadata formats such as HTML meta tags are popular and useful, and have contributed immensely to the spread of metadata tagging. LOM has a relatively complex structure, but it is similar enough to the structure of XML documents to be simple to use. RDF provides no such wide-spread syntax, and the apparently steep learning curve remains

a major obstacle to the acceptance of RDF. For example, while several versions of the RSS news syndication format have tried to use RDF, new versions seem to always step away from RDF in favor of a more predictable XML approach.

This points to a general observation: in any given application it is always easier to devise a custom XML language with custom semantics than to use a complex metadata framework. The extra work involved in being compatible with such a metadata framework does not become evident until the amount of metadata interactions increases beyond a certain threshold. It seems more and more systems are reaching that threshold and are looking for such a framework.

On the other hand, if the RDF specifications are not reused for such a framework, there is a real risk of reinventing much of what has already been achieved within the Semantic Web. Dublin Core is one example, as its abstract model closely resembles that of RDF. Dublin Core on the one hand uses its own abstract model and metadata formats, while on the other hand it relies on RDF Schema to specify the machine-processable semantics of its terms.

The solution envisioned in the framework proposed in this thesis allows for the best of both worlds, by opening up the possibility of retaining tailored XML languages and other metadata standards, while relying on a core model for facilitating harmonization.

Currently, it's hard to predict with certainty if RDF will be the future core model of metadata standards. But it seems certain that many of the features of RDF are destined to become part of future core models, whatever form they will take.

We conclude with two statements:

1. RDF is the only framework in existence today with both the traction and feature set to function as a core model in the sense envisioned here.
2. We have outlined the necessary harmonization features of any future metadata framework that intends to fill the role of RDF for metadata harmonization purposes.

It should be noted that this analysis does not include an evaluation of RDF regarding other aspects, such as tool support, knowledge representation characteristics, web integration etc., but only as a foundation for metadata harmonization. For this reason, this thesis has avoided topics such as ontologies, linked data, databases and tools, etc., and focused solely on the more abstract issues of metadata interoperability and harmonization.

7.6 Implications for Current Metadata Standards

We now turn to a short summary of the direction the above conclusion point towards for the set of specifications studied in this thesis. We will base the analysis on the previous conclusion that the only feasible core model for metadata harmonization today is RDF, together with RDF Schema as the schema model.

7.6.1 The Dublin Core Set of Specifications

Dublin Core already builds heavily on RDF Schema and the RDF model. The Dublin Core terms are specified independently of any particular syntax or application profile, and are therefore at an advanced stage regarding harmonization.

The Dublin Core abstract model and its relationship to RDF has a long and complex history. It currently lacks a clear definition of its abstract syntax and a well-designed semantic embedding to RDF, but these aspects could easily be amended in an updated Dublin Core Abstract Model.

Dublin Core has a long history of developing application profiles, and the work on Description Set Profiles and the Singapore Framework for application profiles are good candidates for generalization to, for example, RDF.

7.6.2 IEEE LOM and the IMS Standards

The priorities for LOM according to this analysis can be divided into two parts:

1. Short-term: produce a LOM RDF vocabulary that can be used to construct a mapping from LOM to RDF. Such a vocabulary is at the late stages of standardization in the IEEE, and a mapping is also underway.
2. Long-term: split LOM into multiple standards with different updating schedules and procedures: an abstract model, a core element vocabulary, value vocabularies and application profiles. It should be noted that a LOM RDF vocabulary is a first important step in this direction.

The same sort of analysis can be made regarding the IMS set of specification.

7.6.3 The Library Metadata Standards: MODS, METS, RDA

It is not reasonable to expect that MODS and METS, being XML-based standards, should be reconstructed to build on an abstract model. Instead we should encourage two things:

1. The definition of an RDF vocabulary for all metadata elements used in the respective specifications
2. The introduction of GRDDL support in the XML formats, providing for automatic translation of the XML into RDF using the above vocabulary.

With these two simple steps, MODS and METS are capable of participating in harmonized metadata activities.

For RDA, the issue is more complex. In theory, RDA builds on the Dublin Core abstract model, using a property-value model to define metadata elements. Or that is at least the intention. In practice, RDA has not been very careful in the metadata modeling, and the element categorization is only described in a working document, not the standard itself[44]. Unfortunately, this documentation is not detailed enough to actually produce an RDF Schema, as evidenced by efforts to produce an official RDA schema (Hillmann et al., 2010). So for RDA, the main efforts regarding harmonization should be spent on:

44 "RDA Element Analysis" as appearing on the RDA web site http://www.rda-jsc.org/working2.html#rda-element

1. Ensuring that the RDF vocabularies for RDA are stabilized and adopted as part of RDA proper.
2. Ensuring that metadata produced using RDA actually adhere to the Dublin Core abstract model, not only in theory but in practice. It is not an unreasonable concern that given the legacy demands on RDA (MARC21 compatibility etc), RDA implementations will not prioritize Dublin Core compatibility despite the prominent place of Dublin Core in the base RDA documentation.

If these two issues can be addressed, RDA is in a position to participate fully in metadata harmonization.

7.6.4 ISO MLR

ISO MLR[45] is based on a property-value model like RDF and Dublin Core. Much like Dublin Core, and unlike LOM, ISO MLR comes with a clear separation between the abstract syntax with semantics, the element vocabularies and application profiles. In this area, ISO MLR closely follows the guidelines developed here.

It is important to understand that a stated goal of MLR is to function as a harmonizing bridge between IEEE LOM, Dublin Core and other metadata standards. Therefore, it is reasonable to compare MLR to the requirements on a core model in our harmonization framework. In that context, MLR has a number of issues :

1. MLR does not use an URI-based identification method, but relies on a custom identification system. This means that MLR cannot usefully reuse other properties. On the other hand, this does not preclude a semantic embedding from MLR to RDF.
2. MLR does specify an informal semantics for element vocabularies, among other things classifying them as classes, properties, etc. MLR does not specify a machine-processable format for element vocabularies.
3. MLR itself does not specify a formal semantics of the abstract model, instead relying on an intuitive understanding of the concepts of properties and classes.

Thus, in comparison with RDF, MLR lacks some of the fundamental components for harmonization. We conclude that MLR is not a desirable basis for harmonization, and an unsuitable target for semantic embeddings.

Instead, MLR needs to make itself semantically embeddable into RDF. Two practical steps are needed:

1. The production of RDF vocabularies for all of the terms defined in MLR.
2. The definition of an official mapping into RDF using the RDF vocabularies.

Both of these steps should be relatively straightforward, but they should be made a high priority for the ISO JTC1 SC36 community.

45 based on the latest drafts at the time of writing

7.6.5 MPEG-7

MPEG-7 is an XML-based standard, firmly rooted in XML Schema technologies, so we expect the developments for MPEG-7 to follow the pattern that we described for MODS and METS above. There is currently no work underway to achieve this, but background work can be found in Hausenblas (2007), van Ossenbruggen et al. (2004) and Nack et al. (2005).

8. Conclusions

We have demonstrated that true metadata interoperability is still, to a large extent, only a vision, and that metadata standards still live in relative isolation from each other. The modularity envisioned in the discussion about application profiles is severely hampered by the differences in abstract models used by the different standards, and efforts to produce vocabularies often end up in the dead end of a single framework. In order to enable automated processing of combined metadata, including extensions and application profiles, the metadata will need to use element vocabularies expressed in a common schema language such as RDF Schema and be made semantically combinable with a common metadata framework such as RDF.

To achieve this, there is a need for a radical restructuring of metadata standards, modularization of metadata vocabularies, and formalization of abstract frameworks. RDF and the Semantic Web provide an inspiring approach to metadata modeling, but it remains to be seen whether that framework will be adopted as a basis for a wide variety of web-oriented metadata standards.

8.1 Contributions of this Thesis

This thesis has focused on designing a theoretical framework for analyzing metadata harmonization issues and providing practical harmonization solutions for current metadata standards based on this framework. We will now summarize the contributions of this thesis to the research questions posed in section 1.3.

8.1.1 Definitions

How can the notions of metadata interoperability and metadata harmonization be meaningfully defined?

Well-grounded and useful definitions of the concepts of metadata, metadata interoperability and metadata harmonization have been developed in section 2. We have shown throughout this thesis how to apply the definitions in a way that leads to a better understanding of the issues. In section 6.4.2 we defined what it means for two metadata standards to be *harmonized*, addressing the core question of this thesis.

8.1.2 Measuring Harmonization

What are the features that determine the level of harmonization between metadata standards, and how can they be measured?

A clear separation of the respective roles of metadata syntax and semantics has been developed in section 4, leading to an understanding of their respective contributions to harmonization issues. We have demonstrated why metadata syntaxes are secondary to harmonization, and the real crux of the problem is the semantics of metadata.

The definition of the notion of an abstract metadata model in section 4.3, coupling abstract syntax and semantics, has been shown to be fundamental in understanding the harmonization issues, in particular between IEEE LOM and Dublin Core. We have introduced the notion of *semantic metadata interoperability* in section 4.4.5 to understand how machine-processable semantics contribute to harmonization.

8.1.3 Harmonization Issues

Where does harmonization fail in currently widely used metadata standards?

We have seen in section 5.5.6 that tools that only work within a given metadata framework, such as application profiles, do not contribute substantially to cross-standard harmonization. In section 6.2 we have analyzed the combinability of metadata fragments and demonstrated how incompatible metadata semantics make such combinations meaningless. Moreover, it has been demonstrated in section 6.3.1 how incompatibilities between abstract models cause the incombinability of metadata fragments from different standards.

A clear separation between issues regarding pre-coordinated (vertical) harmonization and post-coordinated (horizontal) harmonization has been developed in sections 5 and 6, making it substantially easier to isolate the core harmonization issues in horizontal harmonization.

8.1.4 Increasing Harmonization

What are the potential methods of increasing harmonization, and how can they be implemented?

A list of concrete suggestions for increasing harmonization has been presented in section 6.5, and based on the metadata harmonization framework in section 7, a concrete TODO-list has been presented in section 7.6. The presented solutions are based on the concept of *semantic embeddings* defined in section 6.4 and semantic metadata interoperability defined in section 4.4.5.

8.1.5 Harmonization Framework

Can a harmonization framework be formulated that captures the solutions proposed in this thesis?

Section 7 contains a comprehensive model for metadata harmonization that captures the lessons learned in this thesis. The model is based on "separation of concerns" for each specification, thereby increasing the dynamics of metadata standardization processes. The framework can be retrofitted on top of existing standardization activities and it can provide guidance for future developments.

8.2 The Potential in Harmonized Standards

We have shown why harmonized standards are important in order to integrate metadata from different domains. The potential benefits of metadata harmonization are not limited to cross-domain metadata exchange, but extends to several areas of metadata usage.

In Naeve (2005), three stages of metadata development are defined:

1. **Semantic isolation** – when metadata semantics are not compatible. This stage is characterized by syntactic interoperability (e.g. based on XML), and non-networked metadata descriptions. In this stage, metadata interoperability exists mainly in isolated coordinated communities.
2. **Semantic coexistence** – when metadata uses a common semantics, so that descriptions can coexist in the same semantic space. This stage is the stage of basic metadata harmonization, characterized by semantics-aware specifications and networked, graph-based descriptions and interoperability between communities.
3. **Semantic collaboration –** when metadata semantics interact across systems and standards. This stage is characterized by the use of ontology management systems, semantic mappings and controlled evolution of metadata standards.

Reaching the stage of semantic collaboration is a significant development goal beyond metadata harmonization, and harmonization is a prerequisite for this stage.

However, it is often forgotten that much can be gained already from the semantic coexistence stage. In recent years, the Linked Data movement (see Bizer et al., 2009) has demonstrated the value inherent in semantic coexistence based on RDF and the HTTP protocol, without relying on ontologies or other heavy metadata machinery, but rather on taking a light-weight interoperability approach.

Nilsson (2001a) and Nilsson (2001b) presented a set of early lessons from converting the IMS metadata standard to RDF, all of which dealt with the semantic coexistence stage:

- A single storage model that works for combining a multitude of metadata standards. This is the core of what metadata harmonization is about, since a single processing model is a key requirement for harmonization.
- Term reuse between standards is simplified
- Machine-readable relationships between vocabularies
- Machine-readable vocabulary descriptions
- Simple vocabulary extension, and straightforward instance extension.
- Unification of descriptive standardization efforts into a single framework

Similarly, metadata harmonization enables the use of a single abstract query language to span over all metadata used. The underlying potential is made exceptionally clear in the experimental Edutella network – a P2P-based system for querying remote databases using an RDF query language, described in Paper 3 and in Nejdl et al. (2002). In Edutella, there is no coordination of metadata schemas, but the network tries to optimize the routing of queries depending on the metadata terms used, and this is made possible by the use of a harmonized metadata framework.

Examining the potential in large-scale metadata harmonization is thus a broad and exciting topic. We will not further dwell on the potential benefits of ontology-based systems, since this has been studied extensively (see e.g. Ding et al., 2006).

Reaching beyond the semantic collaboration stage, Naeve (2005) shows how semantic collaboration is a prerequisite for building a conceptual interface to the semantic web, making the formally expressed knowledge accessible to humans as well – the *human semantic* web. In general, an important challenge for the future will be how to bring the potential of metadata harmonization to fruit in everyday applications.

8.3 Future Work

This thesis has presented a short-term roadmap for the analyzed metadata standards. A more difficult question is what the medium- and long-term developments will be, both regarding practical issues in standardization and harmonization, and regarding theoretical developments of metadata and semantics.

8.3.1 Stabilizing the Harmonization Framework

We are still far from a situation where metadata harmonization is a natural expectation from metadata standards. The continued proliferation of LOM-based standards is perhaps the clearest sign that isolated models continue to attract attention and resources, as evidenced by new specifications such as IMS LODE and ILOX[46]. In the opinion of the author, these developments are essentially dead ends from a harmonization perspective, even though they solve practical vertical harmonization needs.

Thus, there is a need for a concerted push to overcome some of the chasms building up between the community searching for a common base model for metadata and the various isolated metadata communities such as learning technologies, multimedia and libraries. In short, horizontal harmonization needs to be made part of the functional requirements of future versions of standards such as ISO MLR, IEEE LOM and RDA, and not just an afterthought. This is a political issue that research and technology cannot resolve.

Another important aspect to consider is the value of informal interoperability across standards. One of the reasons Dublin Core has been so successful is that the informal semantics of the Dublin Core properties is available in a very accessible and widely distributed form, for example through the Dublin Core ISO standard (ISO 15836:2009), which presents only the informal semantics of the Dublin Core terms, and which has been widely deployed in a variety of systems that support only a proprietary or custom metadata format. This informal interoperability is an important stepping stone towards more advanced forms of metadata interoperability, as envisioned by the Dublin Core Interoperability Levels document (Nilsson, Baker & Johnston, 2009).

This can be regarded as a practical application-oriented perspective on increasing metadata interoperability, which has not been addressed by this thesis, but which is a potentially very interesting area for future developments and research.

Improved methods for automatically extracting metadata from various formats, resulting in automatic semantic embedding, is an area in strong development. Specifications such as GRDDL for transforming XML languages to RDF, XMP[47] for embedding RDF in PDF files, and RDFa[48] for embedding RDF fragments in HTML are important tools for making RDF metadata ubiquitous and easy to use.

There is one significant gap in the set of specifications available in the proposed framework, assuming that we base it on RDF: a specification of application profiles, as discussed in section 5.5.4. Based on the success of metadata application profiles in both the IEEE LOM and Dublin Core communities, it is surprising that no such technology is in widespread use for RDF. One explanation could be that OWL is viewed as offering all necessary tools, but as explained in section 5.5.4 this is decidedly false. It is a reasonable expectation that interesting developments, possibly based on Dublin Core DSPs, the SHAME editor (Palmér et al., 2007) and Fresnel lenses (Bizer et al., 2005), will be seen in this area.

46 http://www.imsglobal.org/lode/index.html

47 http://www.adobe.com/products/xmp/

48 http://www.w3.org/TR/rdfa-syntax/

8.3.2 Modular Standards, Evolvability and Opportunistic Collaboration

In this thesis we have touched upon a fundamental issue with standards such as LOM, which specifies both an element vocabulary, a set of value vocabularies and an application profile combining these building blocks within a single, monolithic standard. Such a monolithic standard means that even minor revisions to the definition of a single element requires a new revision of the whole standard. For the same reason, a monolithic standard is very difficult to adapt to new technological developments.

By separating the specification of abstract models, the design of application profiles and the declaration of metadata vocabularies, we can reach a partial solution to the differences between the LOM and Dublin Core approaches to application profiles. By using the metadata standardization approach proposed in section 7, these components would be split into separate specifications, leading to a significantly higher incentive for mixing and matching, while still retaining all the advantages of the combined approach in terms of validation and conformance testing.

The harmonization framework proposed in this thesis therefore allows for a more rapid development of metadata standards, since the separate parts can be developed independently without sacrificing interoperability.

Another highly important consequence of a modular harmonization framework based on a common core model is the possibility for post-coordinated collaboration. As the core model specifies a common “grammar” in the form of an abstract model, but leaves the definition of vocabularies to domain specifications, the metadata language has a real chance to evolve dynamically, reusing valuable bits of existing vocabularies and redesigning other parts. RDF is clearly contributing to a metadata standardization ecosystem, where vocabularies compete and collaborate freely on top of the core model.

This follows a general pattern of “disagreement management” (Naeve, 2009, Naeve et al. 2010), where the framework supports resolving vocabulary conflicts bit-by-bit, while allowing for conflicting vocabularies to coexist. From a vocabulary standardization point of view, these features are important tools for creating high-quality vocabulary specifications. One example is the Linked Data community, where vocabulary usage patterns can be discerned, and communities like DCMI are using this information to design new vocabulary, that in turn can be deployed without invalidating or conflicting with existing data. This points to a generalization of the metadata ecosystem notion discussed in Paper 1, from the evolution of metadata descriptions to the evolution of metadata standards.

This notion of “evolvabilty” as a functional requirement for the design of web standards is discussed thoroughly in Berners-Lee (1998). Berners-Lee describes a tension between this requirement and the requirement of interoperability, and our analysis confirms this view. Metadata interoperability is made easier by tightly constrained standards, which on the other hand makes harmonization harder.

8.4 Final Words

The problems in the domain of metadata interoperability and harmonization have crystallized over the last decade, and we can now see significant movement towards consolidation of the accumulated experiences and the implementation of solutions. For example, the developments

within the Dublin Core community in the years 2005-2007 towards a more strongly typed Dublin Core vocabulary, more closely aligned with RDF, were essentially unthinkable just a few years before, according to the author's personal experience.

Similarly, the developments within ISO MLR and RDA are promising in that there is increasing awareness of the need to be “RDF-compatible”, and clear attempts at realizing such compatibility. A major issue has been that the notion of compatibility with RDF has been largely undefined. This thesis has hopefully clarified some of the requirements for “compatibility”, and in particular the notion of semantic embeddings is, in the author's opinion, absolutely fundamental when discussing harmonization issues.

The Linked Data community provides a new and extremely interesting testing ground for metadata deployment. In contrast to much of the earlier Semantic Web work which has had a strong focus on ontologies and formal analysis of metadata, linked data offers a pragmatic, data-oriented environment that showcases the true value of harmonized metadata using hundreds of vocabularies in combination. This is in line with the approach in this thesis, which has been oriented more towards interoperability and harmonization of metadata descriptions than towards formally expressed semantics, for the simple reason that formal semantics requires a high degree of metadata harmonization in order to be useful.

In conclusion, there are interesting times ahead in the metadata standardization business!

Definitions

The following are a set of terms whose definitions have been developed specifically for the purpose of this thesis.

abstract metadata model
: a mapping from an abstract syntax to an interpretation of the syntax as information about a thing. See section 4.3.

abstract syntax
: a specification of the concepts used in a standard, and how they combine to form a metadata description, without reference to a concrete syntax. See section 4.3.

ad-hoc processing
: metadata processing without regard to the machine semantics. See section 4.4.6.

binding
: a specification that defines the encoding of an abstract syntax into a concrete syntax. See section 4.2.1.

element vocabulary
: a set of terms conforming to a metadata standard, and used as the building blocks in metadata instances. See section 6.3.

formal semantics
: a specification of metadata semantics in terms of a formal mathematical model. See section 4.4.

harmonized standards
: a set of metadata standards that can be semantically embedded into another standard. See section 6.4.2.

horizontal harmonization
: interoperability based on interoperability across standards, i.e post-coordination. See section 6.

informal semantics
: a specification of metadata semantics in plain language, intended for human consumption. See section 4.4.

interoperable processing
: metadata processing based on the abstract model and the interoperable semantics. See section 4.4.6.

metadata
: Descriptive data about identifiable things. See section 2.2.3.

machine-processable semantics
: a specification of metadata semantics expressed in a machine-parseable format. See section 4.4.

metadata fragment
: a part of a metadata instance expressed in a concrete or abstract syntax conforming to the structure specified by a metadata standard. See

metadata harmonization
: the ability of two or more systems or components to exchange combined metadata conforming to two or more metadata specifications, and to interpret the metadata that has been exchanged in a way that is consistent with the intentions of the creators of the metadata. See section 2.4.

metadata interoperability
: the ability of two or more systems or components to exchange descriptive data about things, and to interpret the descriptive data that has been exchanged in a way that is consistent with the interpretation of the creator of the data. See section 2.3.

metadata semantics
: an interpretation of a metadata syntax in terms of information about a thing. See section 4.3 and 4.4.

semantic embedding
: a mapping from instances conforming to one metadata standard, to instances of another metadata standard, that preserves the semantics of the metadata instances. See section 6.4.

semantic metadata interoperability
: a situation where two systems can exchange machine-processable semantics alongside the metadata and interpret this semantics correctly. See section 4.4.5.

value vocabulary
: a set of items intended to be used as values of elements in metadata instances. See section 6.3.

vertical harmonization
: interoperability on different levels within a given set of standards, based on pre-coordination. See section 5.

References

Allinson, J., Johnston, P., Powell, A. (2007), A Dublin Core Application Profile for Scholarly Works, *Ariadne* Issue 50, January 2007, http://www.ariadne.ac.uk/issue50/allinson-et-al/

Baca, M. (ed), Gill, T., Gilliland, A. J., Whalen, M., Woodley, M. S. (2008), Introduction to Metadata: Pathways to Digital Information. Online Edition, Version 3.0. http://www.getty.edu/research/conducting_research/standards/intrometadata/

Baker, T. (2003), DCMI Usage Board Review of Application Profiles. http://dublincore.org/usage/documents/profiles/

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

Becket. D. (ed.) (2004), RDF/XML Syntax Specification (Revised), W3C Recommendation 10 February 2004, http://www.w3.org/TR/REC-rdf-syntax/

Becket, D., Berners-Lee, T. (2008), Turtle - Terse RDF Triple Language, W3C Team Submission, http://www.w3.org/TeamSubmission/turtle/

Bearman, D., Miller, E., Rust, G., Trant, J. & Weibel, S. (1999), A Common Model to Support Interoperable Metadata, *D-Lib Magazine*, January 1999. http://www.dlib.org/dlib/january99/bearman/01bearman.html

Berners-Lee (1998), Evolvability, May 2001. http://www.w3.org/DesignIssues/Evolution.html

Berners-Lee, T., Hendler, J., Lassila, O. (2001) The Semantic Web. *Scientific American*, May 2001. http://www.sciam.com/2001/0501issue/0501berners-lee.html

Berners-Lee, T., Fielding, R., and L. Masinter, (2005) "Uniform Resource Identifier (URI): Generic Syntax", RFC 3986, January 2005, http://www.ietf.org/rfc/rfc3986.txt

Berners-Lee, T., Connolly, D. (2008), Notation3 (N3): A readable RDF syntax, W3C Team Submission, http://www.w3.org/TeamSubmission/n3/

Bizer, C., Heath, T., Berners-Lee, T. (2009), Linked data – the story so far, *International Journal on Semantic Web and Information Systems*, 5(3):1–22.

Bizer, C., Lee, R., Pietriga, E. (2005), Fresnel - Display vocabulary for RDF. http://www.w3.org/2005/04/fresnel-info/manual/

Brickley, D. & Guha, R. V. (eds.) (2004), RDF Vocabulary Description Language 1.0: RDF Schema, W3C Recommendation 10 February 2004. http://www.w3.org/TR/rdf-schema/

Cabinet Office (2006), e-Government Metadata Standard Version 3.1, e-Government Unit, London, http://www.cabinetoffice.gov.uk/govtalk/schemasstandards/metadata/egms_31.aspx

Carlyle, A. (2006), Understanding FRBR as a Conceptual Model: FRBR and the Bibliographic Universe, *Library Resources & Technical Services* 50 (2006): 264-73.

Chan, L. M., Zeng, M. L., (2006). Metadata Interoperability and Standardization – A Study of Methodology Part I: Achieving Interoperability at the Schema Level. *D-Lib Magazine*, Volume 12 Number 6. http://www.dlib.org/dlib/june06/chan/06chan.html

Crocker, D. (1982), Standard for ARPA Internet Text Messages, RFC 822, August 1982. http://www.ietf.org/rfc/rfc0822.txt

Cover, R. (1998), XML and Semantic Transparency. http://www.oasis-open.org/cover/xmlAndSemantics.html

Coyle, K., Hillmann, D. (2007). Resource description and access (RDA): Cataloging rules for the 20th century. *D-Lib Magazine*, Volume 13 Number 1/2. http://dlib.org/dlib/january07/coyle/01coyle.html

Day, M. & Cliff, P. (2003), RDN Cataloguing Guidelines, Version 1.1. http://www.rdn.ac.uk/publications/cat-guide/

DCMI Usage Board (2008), DCMI Metadata Terms, DCMI Recommendation. http://dublincore.org/documents/dcmi-terms/

Department for Education and Skills (in association with Simulacra and Schemeta) (2003a), Curriculum Online: Metadata Guide for Tagging, Version 1.11. http://www.curriculumonline.gov.uk/SupplierCentre/Metadataguides.htm

Department for Education and Skills (in association with Simulacra and Schemeta) (2003b), Curriculum Online: A Technical Guide, Version 1.07. http://www.curriculumonline.gov.uk/SupplierCentre/Metadataguides.htm

Ding, L., Kolari, P., Ding, Z., & Avancha, S. (2006). Using Ontologies in the Semantic Web: A Survey. In R. Sharman, R. Kishore, & R. Ramesh (Eds.), *Ontologies: A Handbook of Principles, Concepts and Applications in Information Systems* (pp. 79–114), Berlin: Springer.

Directory Interchange Format (DIF) Writer's Guide, (2009). Global Change Master Directory. National Aeronautics and Space Administration. http://gcmd.nasa.gov/User/difguide/

Dodig-Crnkovic, G., (2010), Constructivist Research and Info-Computational Knowledge Generation, In: Magnani, L.; Carnielli, W.; Pizzi, C. (Eds.) *MODEL-BASED REASONING IN SCIENCE AND TECHNOLOGY Abduction, Logic, and Computational Discovery Series: Studies in Computational Intelligence*, Vol. 314 X, ISBN: 978-3-642-15222-1 Springer, Heidelberg/Berlin, 2010

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. (2001), Metadata Standards, What, Who & Why, *Journal of Universal Computer Science*, 7, (7), 591-601, Springer. http://www.jucs.org/jucs_7_7/metadata_standards_what_who/Duval_E.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

Duval, E. & Hodgins, W. (2003), A LOM Research Agenda. In *Proceedings of WWW2003 - Twelfth International World Wide Web Conference*, 20-24 May 2003, Budapest, Hungary. http://www2003.org/cdrom/papers/alternate/P659/p659-duval.html

Fielding, R. T., Taylor, R. N. (2000). Principled design of the modern Web architecture. In *Proceedings of the 22nd international Conference on are Engineering* (Limerick, Ireland, June 04 - 11, 2000). ICSE '00. ACM, New York, NY, 407-416. http://www.ics.uci.edu/~fielding/pubs/webarch_icse2000.pdf

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

Friesen, N. (2002). Semantic interoperability and communities of practice. *Proceedings of the 11th World Wide Web Conference (WWW2002)*, Hawaii, USA. http://www2002.org/CDROM/alternate/209/

IFLA Study Group on the Functional Requirements for Bibliographic Records (1998), Functional requirements for bibliographic records : final report, München : K.G. Saur, 1998

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

Halpin, H. (2006), Identity, Reference, and Meaning on the Web, *Proc. WWW 2006 Workshop on Identity, Reference, and the Web*, may 2006. http://www.ibiblio.org/hhalpin/irw2006/hhalpin.pdf

Haslhofer, B., Klas, W. (2010), A survey of techniques for achieving metadata interoperability, *ACM Comput. Surv*., vol. 42, no. 2, 2010. http://www.cs.univie.ac.at/upload/550/papers/haslhofer08_acmSur_final.pdf

Hausenblas, M. (ed) (2007), Multimedia Vocabularies on the Semantic Web, W3C Incubator Group Report 24 July 2007, http://www.w3.org/2005/Incubator/mmsem/XGR-vocabularies/

Hayes, P. (ed.) (2004), RDF Semantics, W3C Recommendation 10 February 2004. http://www.w3.org/TR/rdf-mt/

Hazaël-Massieux, D. & Connolly, D., (2005) Gleaning Resource Descriptions from Dialects of Languages, W3C Team Submission 16 May 2005. http://www.w3.org/TeamSubmission/grddl/

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/

Heflin, J. (ed.) (2004), OWL Web Ontology Language – Use Cases and Requirements, W3C Recommendation 10 February 2004. http://www.w3.org/TR/webont-req/

Hillmann, D., Coyle, K., Phipps, J., Dunsire, G. (2010). RDA Vocabularies: Process, Outcome, Use. *D-Lib Magazine*, Volume 16 Number 1/2. http://dlib.org/dlib/january10/hillmann/01hillmann.html

Hillmann, D. I., Phipps, J. (2007). Application profiles: exposing and enforcing metadata quality. In P*roceedings of the 2007 international Conference on Dublin Core and Metadata Applications: Application Profiles: theory and Practice*, Singapore, August 27 - 31, 2007, http://www.dcmipubs.org/ojs/index.php/pubs/article/view/41/20

IEEE Computer Society (2002), IEEE Standard for Learning Object Metadata, IEEE Std 1484.12.1-2002, The Institute of Electrical and Electronics Engineers, Inc., New York, USA.

IEEE Standard Computer Dictionary (1990): A Compilation of IEEE Standard Computer Glossaries. New York, NY

IMS Global Learning Consortium (2004), IMS Meta-data Best Practice Guide for IEEE 1484.12.1-2002 Standard for Learning Object Metadata. http://www.imsglobal.org/metadata/mdv1p3pd/imsmd_bestv1p3pd.html

ISO/IEC 13250 (2003), Information technology - SGML applications - Topic maps, International Organization for Standardization, Geneva, Switzerland

ISO/IEC 15836 (2009), Information and documentation -- The Dublin Core metadata element set, International Organization for Standardization, Geneva, Switzerland

ISO/IEC 15938-2 (2002): MPEG-7 (Multimedia content description interface), Part 2: Description definition language, International Organization for Standardization, Geneva, Switzerland

ISO/IEC 19788 (2009): Metadata for Learning Resources, International Organization for Standardization, Geneva, Switzerland

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

Johnston, P., (2005b), Element Refinement in Dublin Core Metadata, DCMI Recommended Resource, http://dublincore.org/documents/dc-elem-refine/

Johnston, P. (2007), Expressing Dublin Core metadata using the DC-Text format, DCMI Recommended Resource, http://dublincore.org/documents/dc-text/

Johnston, P., Powell, A. (2008), Expressing Dublin Core metadata using HTML/XHTML meta and link elements, DCMI Recommendation, http://dublincore.org/documents/dc-html/

Kasanen, E., Lukka, K. & Siitonen, A. (1993). The Constructive Approach in Management Accounting Research. *Journal of Management Accounting Research*, 5: 241-264.

Klyne, G. & Carroll, J. J. (eds.) (2004), Resource Description Framework (RDF): Concepts and Abstract Syntax, W3C Recommendation 10 February 2004. http://www.w3.org/TR/rdf-concepts/

Lagoze, C,. Van de Sompel, H., Nelson, M., & Warner, S. (2002), The Open Archives Initiative Protocol for Metadata Harvesting, Protocol version 2.0 of 2002-06-14. http://www.openarchives.org/OAI/openarchivesprotocol.html

Lambe, P. (2007), Organising Knowledge: Taxonomies, Knowledge and Organisational effectiveness, Chandos Publishing, Oxford.

Lukka, K. (2003), The constructive research approach. In Ojala, L. & Hilmola, O-P. (eds.) *Case study research in logistics. Publications of the Turku School of Economics and Business Administration*, Series B 1: 2003, p. 83-101. 19 p.

Lytras, M.D., Sicilia, M-A. (2007), Where is the Value of Metadata? *International Journal of Metadata, Semantics and Ontologies* 2(4): 235-241.

Manola, F. & Miller, E. (eds.) (2004), RDF Primer, W3C Recommendation 10 February 2004. http://www.w3.org/TR/rdf-primer/

Manouselis, N., Soto, J., Ebner, H., Palmér, M., Naeve, A., (2008), A Semantic Infrastructure to Support a Federation of Agricultural Learning Repositories, *International Conference on Advanced Learning Technologies* (ICALT), Santander, Spain, 1-5 July 2008, http://kmr.nada.kth.se/papers/SemanticWeb/OrganicEdunet_ICALT08.pdf

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/

Miles, A. J. & Brickley, D. (2005), SKOS Core Guide, W3C Working Draft 10 May 2005. http://www.w3.org/TR/swbp-skos-core-guide

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

Naeve, A., Nilsson, M. (2004), ICT-enhanced Mathematics Education within the Framework of a Knowledge Manifold, Proceedings *of the 10th International Congress of Mathematics Education (ICME)*, Copenhagen, Denmark, July 4-11, 2004. http://kmr.nada.kth.se/papers/MathematicsEducation/ICME2004-ICT-enhanced-math-ed.pdf

Naeve, A., Nilsson, M., Palmér, M., Paulsson, F. (2005), Contributions to a Public e-Learning Platform – Infrastructure, Architecture, Frameworks and Tools, *International Journal of Learning Technology (IJLT)*, Vol 1, No. 3, pp. 352-381, 2005. http://kmr.nada.kth.se/papers/SemanticWeb/Contrib-to-PeLP.pdf

Naeve, A., (2005), The Human Semantic Web – Shifting from Knowledge Push to Knowledge Pull, *International Journal of Semantic Web and Information Systems* (IJSWIS) Vol 1, No. 3, pp. 1-30, July-September 2005. http://kmr.nada.kth.se/papers/SemanticWeb/HSW.pdf

Naeve, A., Palmér, M., Nilsson, M., Paulsson, F., Quick, K., Scott, P. (2006), CoCoFlash: Conzilla, Confolio, and FlashMeeting Integration for Enhanced Professional Learning, *Proceedings of the ICALT-2006 conference*, pp. 1186-1187, Kerkrade, The Netherlands, 5-7 July, 2006. http://kmr.nada.kth.se/papers/ConceptualBrowsing/CoCoFlash-ICALT.pdf

Naeve, A., (2009), Disagreement Management: Increasing the organizational performance of Humanity Inc., Invited talk at the TENCompetence Winter School in Innsbruck, February, 2009, http://www.slideshare.net/EagleBear/ambjrn-on-disagreement-managment-988233

Naeve, A., Ebner, H., Nilsson, M., Palmér, M. (2010), Principles of Disagreement Management, in press

National Information Standards Organization (NISO) (2004), Understanding Metadata, Bethesda, MD, NISO Press. http://www.niso.org/publications/press/UnderstandingMetadata.pdf

National Information Standards Organization (NISO) (2007), A Framework of Guidance for Building Good Digital Collections, 3rd ed., A NISO Recommended Practice, NISO, Baltimore, USA. http://framework.niso.org/

Nejdl. W., Wolf, B., Qu, C., Decker, S., Sintek, M., Naeve, A., Nilsson, M., Palmér, M., Risch, T. (2002), Edutella: A P2P Networking Infrastructure Based on RDF, P*roceedings of the 11th World Wide Web Conference* (WWW2002), Hawaii, USA. http://www2002.org/CDROM/refereed/597/

Nilsson, M., Palmér, M. (1999), Conzilla - Towards a concept browser, Master's Thesis, CID-53, TRITA-NA-D9911, Department of Numerical Analysis and Computer Science, KTH, Stockholm. http://kmr.nada.kth.se/papers/ConceptualBrowsing/cid_53.pdf

Nilsson, M. (2001a), *The Role of RDF in the IMS Family of Specifications*, Draft Whitepaper, IMS Global Learning Consortium, http://kmr.nada.kth.se/papers/SemanticWeb/IMS-RDF-Whitepaper.pdf

Nilsson, M. (2001b), The Semantic Web: How RDF will change learning technology standards, Feature article, Centre for Educational Technology Interoperability Standards (CETIS). http://metadata.cetis.ac.uk/content/20010927172953

Nilsson, M. (2002), Geometric Algebra with Conzilla - Building a Conceptual Web of Mathematics. Master Thesis in Mathematics, Department of Mathematics, KTH, Stockholm. http://kmr.nada.kth.se/papers/MathematicsEducation/GAwithConzilla.pdf

Nilsson, M., Palmér, M. & Naeve, A. (2002), Semantic Web Metadata for e-Learning - Some Architectural Guidelines, *Proceedings of the 11th World Wide Web Conference (WWW2002)*, Hawaii, USA. http://www2002.org/CDROM/alternate/744/

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. (2004), The Edutella P2P Network - Supporting Democratic E-learning and Communities of Practice, in McGreal, R. (ed.) *Online education using learning objects*, Falmer Press, New York, 2004, ISBN 0-415-33512-4. http://kmr.nada.kth.se/papers/SemanticWeb/Edutella-chapter.pdf

Nilsson, M., Naeve, A. (2004), On designing a global infrastructure for content sharing in mathematics education, *Proceedings of the 10:th International Conference on Mathematics Education (ICME)*, Copenhagen, Denmark, July 4-11, 2004. http://kmr.nada.kth.se/papers/MathematicsEducation/ICME2004-On_designing.pdf

Nilsson, M., Johnston, P., Naeve, A., Powell, A. (2006a), The Future of Learning Object Metadata Interoperability, in Koohang A. (ed.) *Learning Objects: Standards, Metadata, Repositories, and LCMS*. http://kmr.nada.kth.se/papers/SemanticWeb/FutureOfLOMI.pdf

Nilsson, M., Mason, J., Naeve, A., Powell, A., Johnston, P., Baker, T., Sutton, S., Hillmann, D. I. (2006b), DCMI Comments on WG4 N0145: Working Draft for ISO/IEC 19788-2 – Metadata for Learning Resources – Part 2: Data Elements, ISO IEC JTC1 SC36 Liaison comment, N1203, http://isotc.iso.org/livelink/livelink?func=ll&objId=4993430&objAction=Open

Nilsson, M., Johnston, P., Naeve, A., Powell, A. (2006c), Towards an Interoperability Framework for Metadata Standards, *Proceedings of the International Conference on Dublin Core and Metadata Applications*, Manzanillo, Colima, Mexico 3 - 6 October 2006, http://dcpapers.dublincore.org/ojs/pubs/article/view/835/831

Nilsson, M., Miles, A. J., Johnston, P., Enoksson, F. (2007), Formalizing Dublin Core Application Profiles - Description Set Profiles and Graph Constraints, in Sicilia M-A., Lytras, M. D. (Eds.): *Metadata and Semantics, Post-proceedings of the 2nd International Conference on Metadata and Semantics Research, MTSR 2007*, Corfu Island in Greece, 1-2 October 2007. Springer 2009, http://kmr.nada.kth.se/papers/SemanticWeb/MTSR07-DSPjournalpaper.pdf

Nilsson, M., Powell, A., Johnston, P., Naeve, A. (2008a), Expressing Dublin Core metadata using the Resource Description Framework (RDF), DCMI Recommendation. http://dublincore.org/documents/dc-rdf/

Nilsson, M., Baker, T., Johnston, P. (2008b), The Singapore Framework for Dublin Core Application Profiles, DCMI Recommended Resource, http://dublincore.org/documents/singapore-framework/

Nilsson, M. (2008c), Description Set Profiles: A constraint language for Dublin Core Application Profiles, DCMI Working Draft, http://dublincore.org/documents/dc-dsp/

Nilsson, M., Baker, T., Johnston, P. (2009), Interoperability Levels for Dublin Core Metadata, DCMI Recommended Resource, http://dublincore.org/documents/interoperability-levels/

Nilsson, M., Naeve, A. (2010), Metadata harmonization: a roadmap for standardization, submitted for publication.

Object Management Group (2006). Meta Object Facility (MOF) Core Specification - Version 2.0. http://www.omg.org/cgi-bin/apps/doc?formal/06-01-01.pdf

Palmér, M., Naeve, A., Nilsson, M. (2001), E-learning in the Semantic Age (PDF), *Proceedings of the 2nd European Web-based Learning Environments Conference (WBLE 2001)*, Lund, Sweden, October 24-26, 2001. http://kmr.nada.kth.se/papers/SemanticWeb/e-Learning-in-The-SA.pdf

Palmér, M., Naeve, A., Paulsson, F. (2004), The SCAM Framework: Helping Semantic Web Applications to Store and Access Metadata, *Proceedings of the European Semantic Web Symposium* (ESWC 2004). Heraklion, Greece, May, 2004, Springer, ISBN 3-540-21999-4, http://kmr.nada.kth.se/papers/SemanticWeb/SCAM-ESWS.pdf

Palmér, M., Naeve, A. (2005), Conzilla – a Conceptual Interface to the Semantic Web, Invited paper at the 13:th International Conference on Conceptual Structures, Kassel, July 18-22, 2005. http://kmr.nada.kth.se/papers/SemanticWeb/Conzilla.pdf

Palmér, M., Enoksson, F., Nilsson, M., Naeve, A. (2007), Annotation Profiles: Configuring Forms to Edit RDF. *Proceedings of the International Conference on Dublin Core and Metadata Applications*, Singapore 28 - 31 August 2007, http://www.dcmipubs.org/ojs/index.php/pubs/article/viewFile/27/2

Pietriga, E., Bizer, C., Karger, K., Lee, R. (2006), Fresnel: A Browser-Independent Presentation Vocabulary for RDF, Proceedings of the 5th International Semantic Web Conference, ISWC 2006, Athens, GA, USA, http://www4.wiwiss.fu-berlin.de/bizer/pub/Fresnel-ISWC2006.pdf

Powell, A., Nilsson, M., Naeve, A., Johnston, P. (2007), DCMI Abstract Model, DCMI Recommendation. http://dublincore.org/documents/abstract-model/

Ratanajaipan, P., Nantajeewarawat, E., Wuwongse, V. (2006), Representing and Reasoning with Application Profiles Based on OWL and OWL/XDD, *Proceedings of the First Asian Semantic Web Conference*, Beijing, China, Lecture Notes in Computer Science, vol.4185, pp. 256–262, 2006.

Sauermann, L., Cyganiak, R. (eds) (2008), Cool URIs for the Semantic Web, W3C Interest Group Note 03 December 2008. http://www.w3.org/TR/cooluris/

Shapiro, E. (1989), The family of concurrent logic programming languages. *ACM Comput. Surv.* 21, 3. http://www.wisdom.weizmann.ac.il/~udi/papers/concurrentlogicprog_89.pdf

Shapiro, E. (1991), Separating concurrent languages with categories of language embeddings, in *Proceedings, 23rd Annual ACM Symposium on Theory of Computing*, pp. 198-208, http://www.wisdom.weizmann.ac.il/~udi/papers/separat_concur_lang.pdf

Sheth, A. (1999), Changing focus on interoperability in information systems: from system, syntax, structure to semantics. *Interoperating Geographic Information Systems* - Kluwer Academic Publishers, Norwell, MA, 47:5--29, January 1999 http://knoesis.wright.edu/library/download/S98-changing.pdf

Tillett, B. (2003), What is FRBR?: A Conceptual Model for the Bibliographic University. Library of Congress Catalog Distribution Service, Washington, DC, 2003. http://www.loc.gov/cds/downloads/FRBR.PDF

Uschold, M. & Gruninger, M. (2002), Creating Semantically Integrated Communities on the World Wide Web, Invited Talk, Semantic Web Workshop, Co-located with WWW 2002, Honolulu, HI, May 7 2002. http://semanticweb2002.aifb.uni-karlsruhe.de/USCHOLD-Hawaii-InvitedTalk2002.pdf

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. http://ieeexplore.ieee.org/iel5/93/29587/01343828.pdf?arnumber=1343828

van Assem, M. (2010), Converting and Integrating Vocabularies for the Semantic Web, PhD thesis at Vrije Universiteit Amsterdam. http://www.cs.vu.nl/~mark/papers/thesis-mfjvanassem.pdf

Weibel, Stuart L. (2009), Dublin Core Metadata Initiative: A Personal History. In *Encyclopedia of Library and Information Science*, Third Edition, ed. Marcia J. Bates and Mary Niles Maack. Boca Raton, Fla.: CRC Press. http://www.oclc.org/research/publications/library/2009/weibel-elis.pdf.

World Wide Web Consortium (2009), OWL 2 Web Ontology Language, Structural Specification and Functional-Style Syntax, W3C Recommendation 27 October 2009, http://www.w3.org/TR/owl-syntax/

Zeng, M. L., Chan, L. M., (2006). Metadata Interoperability and Standardization – A Study of Methodology Part II: Achieving Interoperability at the Record and Repository Levels. *D-Lib Magazine*, Volume 12 Number 6. http://www.dlib.org/dlib/june06/zeng/06zeng.html

Paper summaries

Summary of Paper 1: Semantic Web Meta-data for e-Learning – Some Architectural Guidelines

Year: 2002

Authors: Mikael Nilsson, Matthias Palmér, Ambjörn Naeve

Paper presented at the 11th World Wide Web Conference (WWW2002), Hawaii, USA (2002)

The author contributed sections 1 and 2 to this paper.

This paper presents a critical analysis of the state of the art (in 2002) in metadata interoperability for the e-learning domain. The paper, in section 2, questions several widespread architectural assumptions for metadata, summarized in six points, and presents a set of alternative architectural assumptions.

The criticized architectural assumptions are:

- **Assumption: metadata is objective data about data.**

  Due to the authoritative nature of the element definitions used in many widespread metadata standards, few systems are built with an assumption that metadata can be used to express opinions, comments and other non-authoritative information about resources. By focusing on mechanisms for attribution and for expressing the source of metadata, metadata can be used for subjective information – a critical development to make it useful on the web.

- **Assumption: metadata for a resource is produced only once**

  The assumption of authoritative metadata also tends to create metadata workflows where there is a single source of metadata, producing a final version of the authoritative metadata. The author argues that instead, metadata needs to be handled as a continuous work in progress, where updating and modifying descriptions is a natural part of the meta-data publishing process. The result is a *global metadata eco-system*, a place where meta-data can flourish and cross-fertilize, where it can evolve and be reused in new and unanticipated contexts, and where everyone is allowed to participate.

- **Assumption: metadata must have a logically defined semantics.**

  The author argues against one trend in ontology-based metadata systems, where metadata semantics are defined in extremely fine detail, leading to interoperability issues when two such too strongly specified models meet. Instead, the author argues for using more loosely specified metadata specifications that create patchworks of many small vocabu-

laries, developed in small steps by the communities who need them. This approach places a stronger focus on interaction between metadata specifications rather than mathematically perfect metadata specifications.

- **Assumption: metadata can be described by meta-data documents.**

  The paper references a long-running discussion in the metadata community regarding the value of using XML as a base technology for metadata. The author argues that moving away from document-oriented, top-down approaches like XML is fundamental for enabling subjective metadata in a global metadata ecosystem. Instead, metadata specifications need to focus on a building-block approach like that of RDF, where small metadata fragments can easily be combined into larger metadata graphs.

- **Assumption: metadata is the digital version of library indexing systems.**

  The author criticizes the extremely limited view of metadata as a pure indexing technology, like the traditional library cards. Instead, new uses such as fuller descriptions of material, certification and validation of content, annotations and comments need to be taken into account when designing metadata specifications, as well as new usage patterns where dynamic reuse, recombining and extensions of metadata play a larger role.

- **Assumption: metadata is machine-readable data about data.**

  The author argues that even solving the technical interoperability issues will not be enough. It will also be necessary to use metadata as a conceptual bridge between the structure of the Internet and human users, conceptually enhancing the interaction. A long-term research objective called the Conceptual Web is introduced.

The paper provides the foundation for a new direction in metadata interoperability, and has been a strong influence on later work of the author in this domain. A more dynamic and fragment-oriented view of metadata has been an important component in approaching the harmonization issue.

Summary of Paper 2: The LOM RDF binding – Principles and Implementation

Year: 2003

Authors: Mikael Nilsson, Matthias Palmér, Jan Brase

Paper presented at the Third Annual ARIADNE conference (2003).

The author contributed the main content of this paper, except for sections 4 and 5.

This paper serves as final documentation of work done in the years 2001-2003 on the IEEE LOM and IMS metadata specifications. It documents the efforts to produce a binding of the IEEE LOM metadata standard to RDF, detailing the procedure and principal difficulties encountered.

In section 2, the theoretical differences between the XML binding of IEEE LOM and an RDF binding are presented. The section discusses the importance of metamodels (later usually referred to as abstract models, especially in the context of Dublin Core) and the consequences of adopting a semantic model like RDF for a standard build without consideration for semantics. It also discusses the difference between semantic extensions (through refinements) and structural extensions (by adding distinct structures).

Section 3 presents the details of the binding, explaining the design principles. Among other details, the usage of RDF Schemas, the relationship to Dublin Core, the handling of translations and vocabularies and meta-metadata are discussed in some depth. Sections 4 and 5 shows how the binding successfully allows IEEE LOM to be used in a generic, template-based RDF metadata editor.

The paper concludes that if a large number of idiosyncrasies as part of the translation can be accepted, a translation is still feasible.

The results presented in this paper have been pivotal in understanding the underlying interoperability issues both for IEEE LOM and other specifications. Many of the lessons learned here are generalizable to a large set of metadata interoperability issues, and have therefore strongly influenced later research by the author.

Summary of Paper 3: The Edutella P2P Network – Supporting Democratic E-learning and Communities of Practice

Year: 2004

Authors: Mikael Nilsson

Paper published in McGreal, R. (ed.) Online education using learning objects, Falmer Press, New York, 2004, ISBN 0-415-33512-4.

The author contributed the content of this paper.

This paper presents some lessons from the Edutella Peer-to-peer project. Edutella was designed to enable P2P-based federated metadata search and retrieve for learning objects. The RDF-based system (also described in Nejdl et al., 2002) was designed to be completely agnostic of the actual metadata specifications used within the network, while still being able to route requests based on statistical analysis of the available metadata and performed queries. The paper describes the design goals of the network, based on the metadata subjectivity, a vision of a metadata ecosystem and a fully structurally and semantically extensible metadata environment.

The paper also presents an Edutella-based scenario for a distributed learning activity, in which the possibilities enabled by well-developed and ubiquitous metadata harmonization are explained. Though the technology behind Edutella was not mature for the task it tried to perform, the implementation still serves as a powerful proof of concept regarding the underutilized potential inherent in metadata, obscured by the major remaining metadata harmonization issues.

Summary of Paper 4: Towards an Interoperability Framework for Metadata Standards

Year: 2006

Authors: Mikael Nilsson, Pete Johnston, Ambjörn Naeve, Andy Powell

Paper presented at the International Conference on Dublin Core and Metadata Applications, Manzanillo, Colima, Mexico 3 - 6 October 2006

The author contributed the major parts of the content of this paper.

This paper follows in the footsteps of the previous paper, by using the experiences from the IEEE LOM RDF binding to propose a conceptual metadata framework for metadata, intended to support the development of interoperable metadata standards and applications. The model rests on the fundamental concept of an “abstract model” for metadata, as exemplified by the DCMI Abstract Model, and is based on concepts and ideas that have developed over the years within the Dublin Core Metadata Initiative, but it is designed to function as a general metadata pattern.

The model presented in the paper incorporates the concepts of metadata vocabularies, schemas, formats and application profiles into a single framework that can be used to analyze and compare metadata standards, and aid in the process of harmonization of metadata standards. The paper then uses the proposed model to briefly compare the structures of the Dublin Core metadata specifications and the IEEE LOM standard. Some of the known fundamental differences between the two standards are analyzed in terms of this model, and the paper also presents some important gaps in the current set of Dublin Core metadata specifications.

An important conclusion of the paper is that metadata specifications need to be divided into four different kinds of specifications:

- The over-arching abstract model standard.
- Metadata format specifications.
- Metadata vocabularies.
- Application profiles.

The authors argue that the long-term solution to metadata harmonization is to proceed towards a shared metadata framework. Having all metadata standards expressed using a common abstract model, or at least using compatible abstract models, would greatly increase harmonization in several ways. It would also create a natural separation between the specification of the structure of metadata descriptions and the declaration of metadata terms used within that structure, so that both LOM vocabularies and Dublin Core vocabularies would appear as metadata vocabularies within that one structure. The authors also argue that great care must be taken to ensure that such an abstract model does not conflict with the emerging metadata format for the Web: RDF.

Summary of Paper 5: Formalizing Dublin Core Application Profiles – Description Set Profiles and Graph Constraints

Year: 2007

Authors: Mikael Nilsson, Alistair J. Miles, Pete Johnston, Fredrik Enoksson

Paper published in Sicilia M-A., Lytras, M. D. (Eds.): Metadata and Semantics, Post-proceedings of the 2nd International Conference on Metadata and Semantics Research, MTSR 2007, Corfu Island in Greece, 1-2 October 2007. Springer 2009, ISBN 978-0-387-77744-3

The author contributed the major parts of the content of this paper.

This paper describes a formalization of the notion of Application Profiles as the term is used in the Dublin Core community. The formalization, called Description Set Profiles, defines syntactical constraints (using an XML-based constraint language) on metadata records conforming to the DCMI Abstract Model.

The definition of a formal model for Description Set Profiles marked an important milestone in the evolution of the Dublin Core Metadata Initiative, and was a validation of the DCMI Abstract Model as a concrete foundation for defining machine-processable application profiles.

The formalization described in the paper focuses on only one core aspect of application profiles: the need for syntactically constraining the metadata instances. As described in the Singapore Framework for Dublin Core Application Profiles (Nilsson, Baker & Johnston, 2008b) developed in parallel to the DSP specification, a DSP is part of a documentation package for Dublin Core Application Profiles (DCAPs) containing

- Functional requirements, describing the functions that the application profile is designed to support, as well as functions that are out of scope
- Domain model, defining the basic entities and their relationships using a formal or informal modeling framework.
- Description Set Profile, as described in this paper
- Usage guidelines, describing how to apply the application profile, how he used properties are intended to be used in the application context etc.
- Encoding syntax guidelines, defining application profile-specific syntaxes, if any.

The DSP thus represents the machine-processable parts of a Dublin Core Application Profile.

The paper also discusses how to map this formalism to syntax-specific constraint languages such as XML Schema.

A few initial proofs of the concepts presented in the paper have been realized using DSP for formal documentation of application profiles, using DSPs to configure metadata editors, and using DSPs to generate XML Schemas for validation.

Summary of Paper 6: Metadata Harmonization: a Roadmap for Standardization

Year: 2010

Authors: Mikael Nilsson

Paper submitted for publication

The author contributed the content of this paper.

This paper analyzes a set of current metadata specifications in an attempt at classifying their characteristics and understand their differences.

A special focus of the paper is the compatibility of corresponding features in the respective standards. The potential for harmonization of those features across the standards is discussed in depth, and the different paradigms used for metadata specification are discussed. The paper uses the classification system for metadata specifications developed in Nilsson et al. (2006a) as a basis for feature comparisons.

The paper identifies three major categories of harmonization issues:

- **Conventions**: The different metadata specifications use different methods for identifying and describing metadata elements and terms from value vocabularies.
- **Models**: The specifications differ substantially in how they define metadata records, and in how metadata is structured and processed.
- **Combinations**: Combining elements to form application profiles, and encoding them in syntaxes are both processes that rely heavily on models as well as conventions.

The paper concludes that three components are fundamental in achieving metadata harmonization:

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 (Baker and Dekkers, 2002).
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.

The paper then presents a concrete long-term roadmap for harmonization of the standards, based on Semantic Web frameworks.