1999-12-22

DUBLIN CORE METADATA INITIATIVE WORKING DRAFT

Dublin Core Reference Model 1.0

DCMI Working Draft - 22 December 1999

This Version: <http://www.mailbase.ac.uk/lists/dc-datamodel/files/wd-dcrm-19991222.htm>

Latest Version: <http://www.mailbase.ac.uk/lists/dc-datamodel/files/wd-dcrm-10.htm>

Editors: Renato Iannella <renato@dstc.edu.au>, Eric Miller <emiller@oclc.org >, Paul Miller <lispm@ukoln.ac.uk >

Status of this document

This document is a Working Draft of the Dublin Core Metadata Initiative [DCMI], and is intended to reflect consensus reached within the DataModel Working Group [DCDM]. Publication as a Working Draft does not imply endorsement by the DCMI.

Comments and feedback should be sent to the working group mailing list, dc-datamodel@mailbase.ac.uk, the archives for which may be browsed at <http://www.mailbase.ac.uk/lists/dc-datamodel/> (NOTE, you must be a member of the WG to post messages to the WG) or, alternatively, send your feedback to the Editors of this Working Draft.

Table of Contents

---

1. Introduction

This document expresses the modelling requirements for the Dublin Core Metadata Element Set [DCMES]. The purpose of which is to establish a common framework from which implementations of DCMES can refer and conform. The Dublin Core Reference Model (DCRM) establishes the semantics for interoperability for instances of the DCMES.

The DCRM provides a consistent view of the functional requirements of the DCMES. Conformance to the functional requirements will support interoperability within and across the Dublin Core and other metadata communities.

The DCRM covers two versions of the element set:

- Basic Dublin Core - which supports the current 15 defined elements of the DCMES; and
- Qualified Dublin Core - which supports the addition of qualified extensions of the DCMES for greater expressibility.

The primary focus of the majority of the Dublin Core Metadata Initiative working groups has been in semantic clarification of the DCMES and the identification of common cross-domain qualifiers to support richer descriptive requirements. While this focus has allowed rapid conceptual development free of the constraints imposed by specific syntactic or structural applications, this has led to a certain lack of clarity at times, especially in relation to development and deployment of 'qualification' mechanisms which enrich descriptions in the Dublin Core. It has also made interoperable implementation difficult, as individual implementors have typically developed their own internal mechanisms for actually encoding Dublin Core; mechanisms which are not always compatible with those of their potential collaborators elsewhere.

A formal defintion and systematic representation of 'qualifiers' are a neccessary component for suporting interoperability accross implementors. To faciliate this, the identification of both implicit and explict requirements are helpful in both designing and validating any formal model. The following are a basis of these functional requirements:

1. **Element Identity.** It is important to clearly identify elements defined by particular resource description communities. This is useful both in understanding a particular context ('Title' in the context of DCMES) and to eliminate potential name-collision (eg 'Title' in the DCMES is different than 'Title in the vCard specification).
2. **Internationalization.** The DCMI has identified the requirement for defining the same descriptive concepts in many different languages.
3. **Modularization/Extensibility.** The DCMI recognized early on that there would never be one "true" declarative set of semantics to represent all resource descriptive requirements. As such, the ability to mix semantics defined in various different communities may be necessary for supporting general descriptive practices.
4. **Semantic Refinement.** Many of the Dublin Core elements encompass broad concepts. It is useful to provides the means for semantically refining such elements in order to say something much more specific. (eg the concept 'Illustrator' in the context of CIMI is a refinement of the concept 'Contributor' in the context of the DCMES).
5. **Identification of Encoding Schemes.** The ability to specify a particular encoding scheme is critical to the interpretation of some types of metadata. Dates, for example are written in different formats in different places. Ambiguity can be resolved by identifying the format in which the date is encoded via recommendations for encoding schemes, but a given application could specify an alternate encoding for local purposes.
6. **Specification of Controlled Vocabularies.** Many disciplines use classification schemes, controlled vocabularies or thesauri to describe the content of resources. Creators of metadata need to be able to specify such vocabularies so as to leverage this important class of tools. The Dewey Decimal System, Medical Subject Headings, the Art and Architecture Thesaurus are examples, but there are many others.

7. **Identification of Structured Compound Values.** The use of compound structures as values of metadata is important to many communities, including the use of authority records for authorized versions of the names of subjects, places, corporate entities, and people.

It is important to emphasize that communities having existing qualification vocabularies and encoding schemes will be able to invoke these, enabling such communities to tailor their metadata to the needs of their constituents while remaining within the generalized metadata architecture of the DCMI. Existing capabilities should not be invented a new, but rather the new technology must be able to integrate these important legacy assets and thereby promote the broader acceptance of these valuable tools.

1.1 About this Specification

This document, along with its normative references, includes all the specification necessary for the implementation of interoperable Dublin Core applications.

The following key words are used throughout the document and should be read as interoperability requirements. This specification uses words as defined in [RFC2119] for defining the significance of each particular requirement. These words are:

MUST or MUST NOT
: This word or the adjective "required" means that the item is an absolute requirement of the specification.

SHOULD or SHOULD NOT
: This word or the adjective "recommended" means that there may exist valid reasons in particular circumstances to ignore this item, but the full implications should be understood and the case carefully weighed before choosing a different course.

MAY
: This word or the adjective "optional" means that this item is truly optional. One vendor may choose to include the item because a particular marketplace requires it or because it enhances the product, for example; another vendor may omit the same item.

Examples of Qualifiers used in this document are for demonstration purposes only and should not be interpreted as official qualifiers.

1.2 Deprecated Terms

A number of terms have been used previously to refer to aspects of Dublin Core metadata. These terms are now no longer recommended as they have been superseeded by the vocabulary of this Reference Model document. These terms include, with the new terms in brackets:

- Simple Dublin Core (Basic Dublin Core)
- Unqualified Dublin Core (Basic Dublin Core)
- Complex Dublin Core (Qualified Dublin Core)
- Subelements (Qualifiers)
- Scheme (Vocabulary Value Qualifier or Encoding Value Qualifier)
- Value Component (Component Element)
- Element Qualifier (Refined Element or Component Element)

---

2. Dublin Core Reference Model

The underlying modelling requirement of the DCMES is to support the description of resources. This is realised via statements about those resources. The DCRM is formally represented by:

Resource
: Any object that has identity.

Statement
: Any characteristic of the Resource and the value of that characteristic.

Description
: A collection of one or more Statements about the Resource.

A Resource is described by the collection of Statements. This collection is referred to as a Description. A Description must include at least one Statement to be valid. Within the DCMES context, a Resource will typically be an information or service resource, but may be applied more broadly.

Figure One below graphically represents this core Model. A Description, containing Statements, is directly associated with the Resource they are describing.

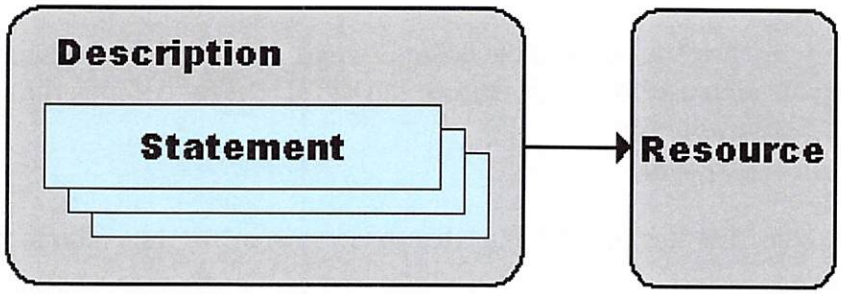

Figure 1 - Basic Dublin Core Reference Model

The DCRM can contain Basic or Qualified Statements. If the DCRM contains Basic Statements, then it is formally known as **Basic Dublin Core**. If the DCRM contains Qualified Statements, then it is formally known as **Qualified Dublin Core**. The DRCM has requirements for **Namespace Identifiers** for the identification of Elements and Qualifiers, and **Extensibility** for additional semantics. DCRM also has requirements for the **Conversion** between Qualified Dublin Core and Basic Dublin Core. These topics are covered in the following sections.

---

3. Basic Dublin Core

Basic Dublin Core exists when the DCRM contains Basic Statements. A Basic Statement consists of an Element and its associated Value. The Basic Statement is formally represented by:

Element
: A characteristic of the Resource. The Core Elements are defined in the [DCMES].

Value
: A String Literal (free text) representing the value of the Element.

The semantics of the Elements have been defined in the Version 1.1 the Dublin Core Metadata Element Set [DCMES]. The Version 1.1 semantics form a normative part of this Dublin Core Reference Model document.

A Basic Statement must include an Element and a Value to be valid.

Figure Two below graphically represents Basic Dublin Core Statements. An Element, represented by one of the DCMES Core Elements, is associated with its corresponding Value, represented by a String Literal.

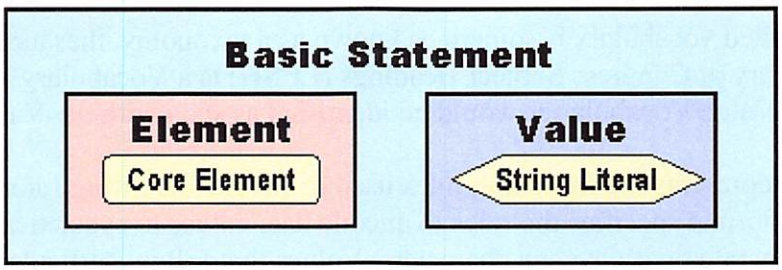

Figure 2 - Basic Dublin Core Statement

Table One below is an example of an instance of Basic Dublin Core metadata for the Resource with identity of <http://dublincore.org/doc/REC-DCRM.html>. The semantics of the Core Elements are defined by [DCMES].

Table 1 - Basic Dublin Core Example

Element	Value
Core Element	**String Literal**
Title	Dublin Core Reference Model
Creator	Renato Iannella
Creator	Eric Miller
Creator	Paul Miller
Date	1999-12-01
Subject	Metadata, Dublin Core, Reference Model

---

4. Qualified Dublin Core

Qualified Dublin Core exists when the DCRM contains Qualified Statements. A Qualified Statement consists of an Element and its associated Value with each being further enunciated with qualifiers.

Qualifiers for the Element are formally represented by:

Refined Element
: A new element that inherits all of the characteristics of the Core Element and may refine some of them. A Refined Element is a specialisation of its more generalised parent Core Element. For example, the Creation Date (specialisation) is a Refined Element for the Date Element (generalisation) since all the characteristics of Creation Date are present in Date.

Component Element
: A new element that groups characteristics of the Core Element into parts. A Component Element is part of of its whole parent Core Element. For example, Time (part) is a Component Element for the Coverage Element (whole) since only temporal characteristics of Coverage are present in Time.

Both Refined and Component Elements may be recursively defined. That is, a Refined Element may be further refined and a Component Element may be further partitioned.

Qualifiers for the Value are formally represented by:

Vocabulary Value

An attribute that represents a controlled vocabulary of terms from where the String Literal was sourced. A controlled vocabulary is sometines known as a taxonomy, thesaurus, ontology or list. For example, the Library of Congress Subject Headings (LCSH) is a Vocabulary Value. Values that follow particular Value Vocabularies would be identified as such with the Value.

Encoding Value

An attribute that represents the formating rules used to encode the String Literal. For example, the W3C Date Time Format specifies the rules to encode date values to syntactically show the year, month, day etc separated with hyphen characters. Values that follow particular Encoding Values would be identified as such with the Value.

Language Value

An attribute that represents the written language of the String Literal. For example, "en" indicates that the Value is written in the English language. The recommended vocabulary for Language Value is defined by [RFC1766]. Note that Language Value is a specialisation of the Vocabulary Value qualifier.

All of the above Element and Value Qualifiers are optional for Qualified Statements. A Qualified Statement must include at least one of the Element or Value Qualifiers to be valid, otherwise it will simply be a Basic Statement. The DCRM does not restrict any of the combinations of various Element and Value Qualifiers, as this will be provided by the semantics for those qualifiers. That is, the Refined Element, Component Element, Vocabulary Value, Encoding Value, and Language Value are not mutually exclusive.

Figure Three below graphically represents Qualified Dublin Core Statements. An Element, represented by one of the DCMES Core Elements, can be further qualified with new Refined and/or Component Elements. A Value, represented by a String Literal, can be further qualifed with Vocabulary, Encoding and/or Language attributes.

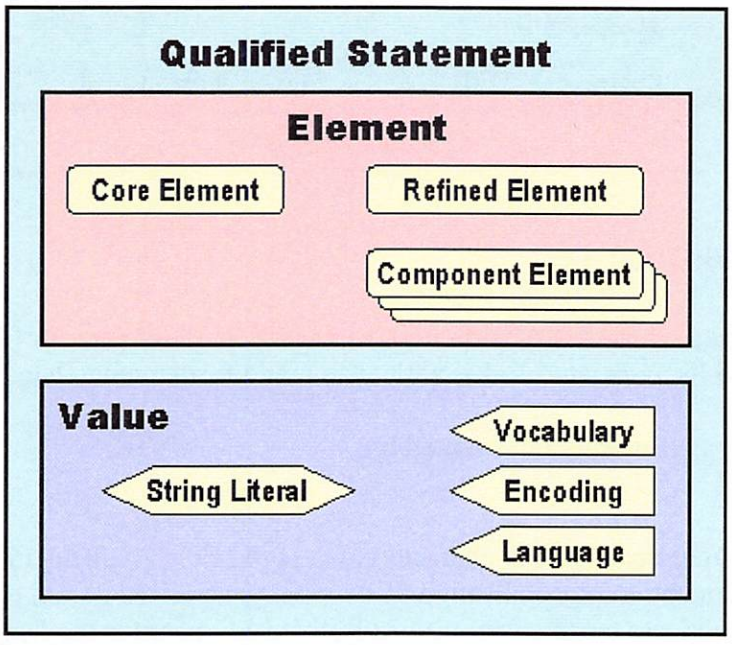

Figure 3 - Qualified Dublin Core Statement

Table Two below is an example of an instance of Qualified Dublin Core metadata for the Resource with identity of <http://dublincore.org/doc/REC-DCRM.html>. The semantics of the Core Elements are defined by [DCMES]. The semantics of the Qualifiers (Refined Element, Component Element, Vocabulary Value, Encoding Value, and Language Value) are provided by DCMI.

Table 2 - Qualified Dublin Core Example

Element			Value			
Core	**Refined**	**Component**	**String Literal**	**Vocabulary**	**Encoding**	**Language**
Title			Dublin Core Reference Model			en
			Modello Di Riferimento Di Dublin Core			it
Creator		Name	Renato Iannella			en
		Affiliation	DSTC			en
Creator		Name	Eric Miller			en
		Affiliation	OCLC			en
Creator		Name	Paul Miller			en
		Affiliation	UKOLN			en
Date	Created		1999-12-01		W3CDT	
	Modified		1999-12-10		W3CDT	
Subject			Metadata; Dublin Core; Reference Model		Keywords	en
			Metadata	LCSH		en

---

5. Namespace Identifier

The semantics of both the Dublin Core Elements and the Qualfiers must be be clearly expressed in all instances of Dublin Core metadata. A unique identifier will be used for this purpose and called the **Namespace Identifier**. The identifier will take the form of a [URI] but no assumptions should be made about this identifier. It should be treated as an atomic identifier.

The Namespace Identifier for Version 1.1 of DCMES will be:

http://dublincore.org/dcb/1.1/#

and will be updated as new versions of the semantics is released.

The Namespace Identifier for Version 1.0 of Dublin Core Qualifiers will be:

http://dublincore.org/dcq/1.0/#

and will be updated as new versions of the semantics is released.

Additionally, a Namespace Identifier can be used at a greater level of granularity to enable the unique

identification of particular elements and qualifiers within the above two sets of semantics.

---

6. Extensibility

The DCMES is built on the principle of extensibility. Other communities can add new Elements and additional Qualifiers that meet their particular needs. In such cases, a new Metadata Element Set is produced that is *based on* the DCMES. The additional semantics (Elements and Qualifiers) should follow the general classification of Qualifiers presented for the Qualified Dublin Core and must be clearly identified with a new Namespace Identifier.

Additionally, the new semantics must follow the principles outlined in this Reference Model. That is, new Element Qualifiers must refine or partition the Core Elements and new Value Qualifiers must represent additional Vocabulary, Encoding, or Language attributes. Additional Vocabulaires for the Language Value qualifier (apart from RFC1766) are permissable as long as the source Vocabulary is always identified.

---

7. Conversion Between Qualified and Basic

In a heterogeneous environment such as that in which the Dublin Core operates, it is necessary to offer a means by which perceived over-complexity may be handled. Such a need is manifested in cases where users or software expecting Basic Dublin Core encounter Qualified Dublin Core, as well as in situations where users or software able to handle Qualified Dublin Core encounter terminology from additional Namespaces with which they are unfamiliar, and which they do not wish to resolve through reference to the Namespace declaration itself.

In both cases, a clear and unambiguous mechanism is required by which the acceptable values (string literals in the first example, and values only from understood Namespaces in the second) may be extracted from the description in such a way that intelligible results are obtained without significant loss of value.

The core principle for conversion is that when all qualifiers, including extensions, are removed from a Qualified Dublin Core metadata record, the resultant Basic Dublin Core metadata record will still makes semantic sense. That is, the information is not misleading. For example, Component Elements must be appended to (re)create an appropriate value for the Core Element.

Also, every validly expressed Basic Dublin Core record is considered to be a valid description in Qualified Dublin Core.

---

8. Conformance Requirements

Proposals for syntactical implementations of the DCMES must address the Dublin Core Reference Model functional requirements presented in this document. The proposals must include clear and unambiguous instructions on how the conformance requirements have been addressed. Dublin Core Element Set implementation proposals should submit their proposals to the DCMI if official conformance is required.

This requirements document may be updated in the future to reflect changes or additional versions of the Dublin Core Reference Model.

The proposals must address the conformance points outlined in the following sections.

8.1. Basic Conformance

The Basic Dublin Core Reference Model conformance requirements for interoperable metadata include:

1. Must support the ability to express all of the semantics for Version 1 releases of the Dublin Core Metadata Element Set.
2. Must support the identity of the Core Elements.
3. Must support the ability to express Basic Statements.
4. Must support the ability to group Basic Statements into Descriptions
5. Should support the ability to associate Descriptions with the Resource being described.
6. All Elements must be repeatable.
7. All Elements may be optional.
8. Any constraints on String Literals may be set and must be clearly indicated.

8.2. Qualified Conformance

The Qualified Dublin Core Reference Model conformance requirements for interoperable metadata include:

1. The Basic Conformance requirements must be meet.
2. Must support the semantics for all Version 1 releases of the Dublin Core Qualifiers.
3. Mechanisms to support and identify Element Qualifiers must be addressed.
4. Mechanisms to support and identify Value Qualifiers must be addressed.
5. All Qualifiers must be repeatable.
6. All Qualifiers may be optional.
7. Support for RFC1766 for the Value Language qualifier must be supported.
8. Mechanisms to associate Qualifiers with the relevant Element must be supported.
9. Mechanisms to allow Elements (including Qualifiers) to be grouped together must be supported.
10. Mechanisms to allow the ordering of Elements (including Qualifiers) may be addressed.

8.3. Namespace Conformance

The Namespace identifier conformance requirements for interoperable metadata include:

1. The Namespace Identifier referring to the appropriate Element and Qualifier semantics must appear.
2. The Namespace Identifier may be used to uniquely identify and refer to particular Elements and Qualifiers.

8.4. Extensibility Conformance

The Extensibility conformance requirements for interoperable metadata include:

1. The Namespace Identifier referring to the new Elements and Qualifier semantics must appear.
2. The new Elements and Qualifiers must be associated with a new Namespace Identifier.
3. New Qualifiers (for DCMES Elements) must meet the same semantic principles as for the current DCMES qualifiers.

8.5. Conversion Conformance

The Conversion conformance requirements for interoperable metadata include:

1. Qualified Dublin Core must be able to be converted to Basic Dublin Core without any negative semantic side-effects.
2. Qualifiers must be able to be removed from Qualified Dublin Core to create a Basic Dublin Core Description
3. Component Elements must be able to be appended to form the Value for a Core Element.
4. Addtional information must not be added to Basic Dublin Core when being converted to Qualified

Dublin Core.

5. Basic and Qualified Dublin Core must be able to ingore extensions, such as additional Elements and Qualifiers, and result in valid Dublin Core records.

---

9. References

[DCDM] Dublin Core DataModel Working Group
<http://purl.org/DC/groups/datamodel.htm>

[DCMES] Dublin Core Metadata Element Set: Reference Description Version 1.1
<http://purl.org/DC/documents/rec-dces-19990702.htm>

[DCMI] Dublin Core Metadata Initiative Home Page
<http://purl.org/dc/>

[RFC1766] Tags for the Identification of Languages, IETF RFC 1766
<http://www.ietf.org/rfc/rfc1766.txt>

[RFC2119] Key words for use in RFCs to Indicate Requirement Levels, IETF RFC 2119
<http://www.ietf.org/rfc/rfc2119.txt>

[URI] Uniform Resource Identifiers (URI): Generic Syntax, Internet RFC 2396
<http://www.ietf.org/rfc/rfc2396.txt>

---

10. Acknowledgements

The Editors wich to acknowledge all the members of the DataModel Working Group who helped to develop the ideas expressed here and apologise if any of their inputs have been misrepresented in this document.
