Topic: Agenda for Usage Board meeting, Washington DC, 19-20 May 2005
Identifier: http://www.bi.fhg.de/People/Thomas.Baker/agenda/
Created: 2005-05-13
Modified: 2005-05-13 10:52, Friday
Maintainer: Tom Baker

2005-05-19: 10:00-12:00 (2 hours)

01. Accessibility [Tom]
http://www.bi.fhg.de/People/Thomas.Baker/agenda/topic-accessibility

2005-05-19: 13:00-17:30 (4 hours)

02. DCMI Extension Namespaces and review of application profiles [Tom]
http://www.bi.fhg.de/People/Thomas.Baker/agenda/topic-profiles

03. Review of Collection Description Application Profile [all]
http://www.bi.fhg.de/People/Thomas.Baker/agenda/topic-profiles-collection

04. Review of Library Application Profile [Tom]
http://www.bi.fhg.de/People/Thomas.Baker/agenda/topic-profiles-libraries

2005-05-20: 9:00-12:00 (2+ hours)

05. MARC relator terms - 1.5 hours [Rebecca]
http://www.bi.fhg.de/People/Thomas.Baker/agenda/topic-relators

06. dc:language - 30 min [Rebecca]
http://www.bi.fhg.de/People/Thomas.Baker/agenda/topic-language-comment

07. Type vocabulary - 1 hr [Stuart]
http://www.bi.fhg.de/People/Thomas.Baker/agenda/topic-type-vocabulary

2005-05-20: 13:00-17:30 (4 hours)

08. Process Document - 30-45 min [Stuart, Diane]
http://www.bi.fhg.de/People/Thomas.Baker/agenda/topic-process

09. Using Dublin Core and AskDCMI - 30 min [Diane]
http://www.bi.fhg.de/People/Thomas.Baker/agenda/topic-userguide

10. Madrid meeting [Tom]
http://www.bi.fhg.de/People/Thomas.Baker/agenda/topic-madrid

11. Definitions and Labels - 1 hour? - or Madrid... [Andy]
http://www.bi.fhg.de/People/Thomas.Baker/agenda/topic-definitions-labels

12. Encoding scheme types - 1 hour? - or Madrid... [Andy]
http://www.bi.fhg.de/People/Thomas.Baker/agenda/topic-encoding-scheme-types

Topic: Accessibility
Identifier: http://www.bi.fhg.de/People/Thomas.Baker/agenda/topic-accessibility/
See also: http://www.bi.fhg.de/People/Thomas.Baker/agenda/
Created: 2005-05-11
Modified: 2005-05-13 10:52, Friday
Maintainer: Tom Baker

In Shanghai we decided to approve "Accessibility" as an element
but for various reasons were unable to finalize the decision.

In Washington, we need to either approve the term, or to
decide to send it back to the Working Group. If we do the
latter, we should be able to formulate a position on why we
were unable to finalize the decision.

As background, everyone should please review the following materials:

-- 2005-04-13: Summary of discussion
http://www.bi.fhg.de/People/Thomas.Baker/public/2005-04-13.accessibility.txt

-- 2005-04-13: Digest of some list discussion
http://www.bi.fhg.de/People/Thomas.Baker/public/2005-04-11.accessibility.txt

-- 2005-04-18: Liddy's reaction
http://www.bi.fhg.de/People/Thomas.Baker/public/2005-04-18.accessibility.txt

-- 2005-04-21: Stu's opinion
http://www.bi.fhg.de/People/Thomas.Baker/public/2005-04-21.accessibility.txt

Here is the revised proposal from Liddy as of 2005-05-13:

-- Cover page:
http://www.ozewai.org/DC-term-proposal/index.html

-- Background material:
http://www.ozewai.org/DC-term-proposal/criteria2.html
http://www.ozewai.org/DC-term-proposal/decision-tree2.html
http://www.ozewai.org/DC-term-proposal/overview2.html

-- The proposal - what we need either to approve or to send back to the WG:
http://www.ozewai.org/DC-term-proposal/prop-reqs-table2.html

-- Examples:
http://www.ozewai.org/DC-term-proposal/examples.html

Topic: DCMI Extension Namespaces and Review of Application Profiles
Identifier: http://www.bi.fhg.de/People/Thomas.Baker/agenda/topic-profiles/
See also: http://www.bi.fhg.de/People/Thomas.Baker/agenda/
Created: 2005-05-12
Modified: 2005-05-13 10:52, Friday
Maintainer: Tom Baker

In Washington, we need to clarify -- in terms of policy, process, and principle -- the context within which the Usage Board will participate in the creation and maintenance of DCMI Extension Namespaces (EXT-NSes).

According to the proposal [1], terms would normally be put into a EXT-NS by being part of an Application Profile (DCAP) developed by a DCMI Strategic Activity and put forward for review by the Usage Board. If a DCAP were found by the Usage Board to be "conformant" with the DCMI Abstract Model, any terms in that DCAP which had not yet been declared could be put into an EXT-NS. By definition, a DCMI Strategic Activity would be sponsored by an organization willing to share maintenance responsibility for the terms with DCMI, and the commitment of DCMI to maintain terms in an EXT-NS over the long term would be subject to declared limitations and processes.

We should discuss, if only briefly, some of the open issues with regard to the implementation of EXT-NSes. These issues, elaborated in [2], include:

-- Use of the term "namespace" (Namespace Policy and Abstract Model)
-- Naming issues: use of acronyms or date-stamps in URI strings
-- Batching of terms into separate DCMI Extension Namespaces
-- Scope of EXT-NSes: to include controlled vocabularies?
-- DCMI maintenance commitment and responsibility
-- Maintenance role of outside organizations

More importantly, we need to clarify by what processes and criteria we will undertake the review of DCAPs by which terms will be put into an EXT-NS. In order to do this, we should all read (or skim):

-- The EXT-NS proposal [1] and issues list [2]
-- The DCMI Namespace Policy, which will need to be extended [3]
-- Relevant Usage Board process documents [4,5,6]
-- Expectations as to the documentational format of DCAPs [7]
-- Naming and identification of terms in an EXT-NS [8,9]
-- The Abstract Model [10] and clarifications thereof [14]
-- In particular, usability of XML elements as DCMI elements [11,12,13]
-- Prior experience in the UB on reviewing DCAPs [15]

After discussing these issues in general, we will look specifically at the DCAPs being developed by the Collection Description and Libraries working groups in order to determine, in more detail, how such a review would work in practice.

[1] DCMI Extension Namespaces proposal
http://www.bi.fhg.de/People/Thomas.Baker/public/2005-05-04.DCMI_Extension_Namespaces.txt
[2] Issues list
http://www.bi.fhg.de/People/Thomas.Baker/public/2005-05-13.extns-issues.txt
[3] DCMI Namespace Policy -> in supplementary packet
http: //dublincore.org/documents/dcmi-namespace/
[4] DCMI Usage Board Review of Application Profiles
http://dublincore.org/usage/documents/profiles
[5] DCMI Usage Board Process: "Proposals for Registration of Application Profiles"
http://www.bi.fhg.de/People/Thomas.Baker/public/2005-05-13.profile-review.txt
[6] Procedure for Approval of DCMI Metadata Terms and Recommendations

http://dublincore.org/usage/documents/approval/
[7] Dublin Core Application Profile Guidelines -> in supplementary packet
http: //dublincore.org/usage/meetings/2004/03/cwa14855-20040210.pdf
[8] DCMI Policy on Naming Terms -> in supplementary packet
http: //dublincore.org/documents/naming-policy/
[9] Guidelines for Assigning Identifiers to Metadata Terms -> in supplementary packet
http: //www.ukoln.ac.uk/metadata/dcmi/term-identifier-guidelines/
[10] DCMI Abstract Model (DCMI Recommendation) -> in supplementary packet
http: //dublincore.org/documents/abstract-model/
[11] XML, RDF, and DCAPs (no status yet)
http://www.ukoln.ac.uk/metadata/dcmi/dc-elem-prop/
[12] DCMI Mixing and Matching FAQ (first draft)
http://www.ukoln.ac.uk/metadata/dcmi/mixing-matching-faq/
[13] DC-Libraries and DC-Architecture discussion about using XML elements
http://www.bi.fhg.de/People/Thomas.Baker/public/2005-02-15.dc-architecture-digest.txt
[14] Element Refinement in Dublin Core Metadata (draft DCMI Recommended Resource)
http://www.ukoln.ac.uk/metadata/dcmi/dc-elem-refine/
[15] Review of PBCore, June 2004
http://www.bi.fhg.de/People/Thomas.Baker/public/2004-06-22.PBCore-final.txt

Topic: Collection Description Application Profile
Identifier: http://www.bi.fhg.de/People/Thomas.Baker/agenda/topic-profiles-collection/
See also: http://www.bi.fhg.de/People/Thomas.Baker/agenda/
Created: 2005-05-11
Modified: 2005-05-13 10:52, Friday
Maintainer: Tom Baker

The Collection Description Application Profile will be the first full formal review by the Usage Board of an application profile, so in Washington we should aim at:

-- defining which aspects of the profile we will review and with what criteria;
-- who can shepherd which aspects of the review;
-- what interaction with the working group, and with what deadlines, will be necessary in order to get the profile onto the agenda in September.

To prepare, we should all read:

-- the "summary" version of the profile [1]. I have included the full version in the packet as reference [2]. (Note: QNames with shaded background are still in need of URIs.)

-- Pete's summary of open or unresolved issues with regard to the draft profile [3].

-- collection-description-related terms approved by the Usage Board in 2004 [4]. (Note the reference in this decision text to "the ambiguity inherent in the existing usage of dc:identifier"!)

-- collection-description-related terms approved by the Usage Board in Shanghai which have not yet been announced [5], as these terms could potentially be given URIs in a DCMI Extension Namespace. We should discuss in Madrid any advantages or disadvantages for doing so.

-- In theory, an DCMI Extension Namespace would provide a place to declare (i.e. assign URIs to) the terms in controlled vocabularies. Finding a home for these vocabularies has in fact been a crucial obstacle to the finalization of the Collection Description AP. The vocabularies in question are included here [6,7,8,9].

-- We should be aware that the WG is considering whether terms would need to be declared as skos:ConceptSchemes in addition to being declared as "classes of concepts" (Vocabulary Encoding Scheme). This would mean more URIs.

[1] http://www.ukoln.ac.uk/metadata/dcmi/collection-ap-summary/
[2] http://www.ukoln.ac.uk/metadata/dcmi/collection-application-profile/
[3] http://www.bi.fhg.de/People/Thomas.Baker/public/2005-05-11.dccdap-issues.txt
[4] http://dublincore.org/usage/decisions/2004/2004-02.Collection-terms.shtml
[5] http://www.bi.fhg.de/People/Thomas.Baker/public/2005-03.Collection-terms.shtml
[6] http://www.ukoln.ac.uk/metadata/dcmi/collection-type/
[7] http://www.ukoln.ac.uk/metadata/dcmi/collection-DCCDAccrualMethod/
[8] http://www.ukoln.ac.uk/metadata/dcmi/collection-DCCDAccrualPeriodicity/
[9] http://www.ukoln.ac.uk/metadata/dcmi/collection-DCCDAccrualPolicy/

Topic: Libraries Application Profile
Identifier: http://www.bi.fhg.de/People/Thomas.Baker/agenda/topic-profiles-libraries/
See also: http://www.bi.fhg.de/People/Thomas.Baker/agenda/
Created: 2005-05-11
Modified: 2005-05-13 10:52, Friday
Maintainer: Tom Baker

Read the latest version of the application profile:
-- http://dublincore.org/documents/2004/09/10/library-application-profile/

Robina reports that the profile has not been updated since September 2004 (i.e. before the Usage Board decisions in Shanghai). She has an action to update it in line with those decisions but has had an email discussion with Rebecca in the meantime. They decided it was not a good use of time to update it while the main issues outstanding are still unresolved. These issues are:

-- the desire to use the Roles from the MARC Relator list LoC has worked on. What is best practice for including entire sets of possible refinements in a DCAP? There are too many MARC relator terms to cite them in the DCAP documentation, and the set of MARC relator terms will change over time. Can the set of MARC relator terms be cited as a whole?

-- the desire to include a couple of terms from the MODS namespace and the mismatch between a DC element and an XML attribute. At any rate, it was the mismatch in the data models that made this kind of re-use problematic.

-- How to describe/define encoding schemes in the AP. In the AP, encoding schemes are referred to in the table describing the element it qualifies and then there is a section at the end of the AP with a table per encoding scheme, following the model Pete uses.

Robina writes: "DC-Lib is a true application profile - it does not declare any terms of its own, only reuses those from other namespaces. So we do not have the problem of where to keep terms we have coined ourselves and InfoURI seems to offer a way forward for encoding schemes and controlled vocabs. However, so far as I know, no-one actually uses DC-Lib as it stands. TEL used it as a basis for the profile it developed and we at the BL have done likewise for a BL profile. We (BL) have coined some terms of our own and are beginning to think about establishing a namespace to keep them in."

Topic:	MARC Relator terms
Identifier:	http://www.bi.fhg.de/People/Thomas.Baker/agenda/topic-relators/
See also:	http://www.bi.fhg.de/People/Thomas.Baker/agenda/
Created:	2005-05-13
Modified:	2005-05-13 10:52, Friday
Maintainer:	Tom Baker

This topic will be issued as a separate packet on Monday, 12 May.

Topic: Comment for dc:language
Identifier: http://www.bi.fhg.de/People/Thomas.Baker/agenda/topic-language-comment/
See also: http://www.bi.fhg.de/People/Thomas.Baker/agenda/
Created: 2005-05-11
Modified: 2005-05-13 10:52, Friday
Maintainer: Tom Baker

In Washington, we should vote on the change proposed below by Rebecca.

The problem

Martin Dürst <duerst@w3.org> has pointed out that the comment for the element "Language" currently says:

> Recommended best practice is to use RFC 3066 [RFC3066] which, in conjunction with ISO639 [ISO639]), defines two- and three-letter primary language tags with optional subtags. Examples include "en" or "eng" for English, "akk" for Akkadian", and "en-GB" for English used in the United Kingdom.

He recommends that this be fixed on the grounds that "eng" is not valid in RFC 3066. He says that RFC 3066 clearly says that if there is a two-letter and a three-letter code for a language, the two-letter code MUST be used.

The relevant passage in RFC 3066 (http: //www.ietf.org/rfc/rfc3066.txt) is point 2 under section 2.3:

> 2.3 Choice of language tag
>
> One may occasionally be faced with several possible tags for the same body of text.
>
> Interoperability is best served if all users send the same tag, and use the same tag for the same language for all documents. If an application has requirements that make the rules here inapplicable, the application protocol specification MUST specify how the procedure varies from the one given here.
>
> The text below is based on the set of tags known to the tagging entity.
>
> 1. Use the most precise tagging known to the sender that can be ascertained and is useful within the application context.
>
> 2. When a language has both an ISO 639-1 2-character code and an ISO 639-2 3-character code, you MUST use the tag derived from the ISO 639-1 2-character code.

The Proposal

Rebecca proposes that we just give an example of a language that has a 3-character code and no 2-character code. So we could change the comment to read:

> Recommended best practice is to use RFC 3066 .... Examples include "en" for English or "ban" for Balinese.

Topic: Type Vocabulary
Identifier: http://www.bi.fhg.de/People/Thomas.Baker/agenda/topic-type-vocabulary/
See also: http://www.bi.fhg.de/People/Thomas.Baker/agenda/
Created: 2005-05-11
Modified: 2005-05-13 10:52, Friday
Maintainer: Tom Baker

In Washington, we should discuss and approve changes in the Type Vocabulary [1]. Essentially, non-definitional text was moved from the definitions into the Comment.

In Shanghai, we decided to sign off on the changes after "editorial consistencies" were corrected. There was some discussion on the list about wordings that we should review [2].

For comparison, the version of the Type Vocabulary currently posted is included in the packet [3].

[1] http://www.bi.fhg.de/People/Thomas.Baker/public/2005-05-02.Type-Vocabulary.html
[2] http://www.bi.fhg.de/People/Thomas.Baker/public/2005-05-04.typevocabulary.txt
[3] http://dublincore.org/documents/dcmi-type-vocabulary/

Topic: Ongoing Revision of the DCMI Process document
Identifier: http://www.bi.fhg.de/People/Thomas.Baker/agenda/topic-process/
See also: http://www.bi.fhg.de/People/Thomas.Baker/agenda/
Created: 2005-05-11
Modified: 2005-05-13 10:52, Friday
Maintainer: Tom Baker

Shepherds: Diane and Stuart

On 2005-05-05, Diane and Stuart agreed to make an interim update of the Process document for (brief) discussion in Washington. This update would cover some of the pending changes, recognizing that the DCMI Extension Namespaces and DCAPs will entail further changes.

This interim update can be found at:
-- http://www.bi.fhg.de/People/Thomas.Baker/public/UB_Process-05-02-05.html

Changes reflected in the above draft include:

- -- [SEE new Preface] Add wording to the UB Process document referring to the DCMI Mission Statement, e.g. to section 4.3 onwards.

- -- [SEE 4.3.1.] Add wording to 4.3.1 of UB Process document to indicate that the criteria are not 'exclusive'

- -- [SEE 4.5.6.3.] Agreed that the announcement of the start of the comment period should come from the shepherd and should say that comments can go to either the appropriate WG mailing list, the dc-general mailing list or privately to the shepherd and should explicitly ask for indications of support for the proposal. [SEE 4.5.6.1.1. & 4.5.6.1.2.] Need to update the UB Process document accordingly.

- -- [SEE 4.5.6.2.] Proposals for new terms should be moved to the DCMI Web site, and given DCMI page headers and a status of 'Proposed term'. when they are accepted by the UB (i.e. before comment period starts).

- -- [SEE 4.7.1.3.] Add wording somewhere in UB Process document to indicate that the recording of decisions should be rich enough that the rationale for decisions is clear to others.

Changes to be done at a later date:

- -- Revisit 3.2 of UB Process document at some stage in light of Scope Statement (see Andrew's action).

- -- Agreed various changes to the UB Process document based on the email from Tom in the meeting packet (Stuart has detailed record of changes required).

- -- Add a new status of 'Endorsed' (for LoC statements about MARC Relator terms) and document processes related to endorsement of such assertions.

- -- Clarify use of the status "Registered" now that UB is no longer accepting proposals for Vocabulary Encoding Schemes.

- -- Need a process for handling changes to definitions such as that contemplated for Date.

- -- Note: Need to consider future possibilities for giving a

status of 'Conforming' to terms in external namespaces.

-- Approval Process Overview: The document 'Procedure for approval of DCMI Metadata Terms and Recommendations' is maintained by the Managing Director as a high-level summary aimed at a general audience

-- http://dublincore.org/usage/documents/approval/ is a one-stop source of the overall procedure. This document links to the more detailed UB process document. The UB process document should now say that the UB will inform Managing Director every time there is a major revision of UB process so that the Managing Director can keep the Approval document in sync.

-- Endorsement mechanism for non-DCMI encoding schemes. We expect that the NLM proposal will be the last proposal to assign a DCMI URI to an external vocabulary. In the medium term, we need to develop a policy, process, mechanism and documentation for endorsing non-DCMI encoding scheme URIs (same endorsement mechanism as for endorsing non-DCMI properties) and develop guidelines to help external organisations/people declare URIs for non-DCMI vocabulary encoding schemes. ACTION (medium-term): Stuart, Diane, Tom, Pete (for RDF mechanism).

Note: Andy's "Guidelines for Assigning Identifiers to Metadata Terms" is reproduced in this packet under Topic 2.

-- Statement of DCMI approach to endorsing vocabulary encoding schemes. We need a public statement of the new approach. ACTION: Tom to formulate short statement for discussion on UB list and in telecons.

-- Longer-term, we need to clarify the role of the proposer as a participant in the Usage Board meeting. They can and should serve as a resource for explaining the proposal, but it has been suggested that we consider asking participants to step out for the discussion.

Topic: Using Dublin Core
Identifier: http://www.bi.fhg.de/People/Thomas.Baker/agenda/topic-userguide/
See also: http://www.bi.fhg.de/People/Thomas.Baker/agenda/
Created: 2005-05-12
Modified: 2005-05-13 10:52, Friday
Maintainer: Tom Baker

For Washington, we should review several revised pages of "Using Dublin Core":

-- http://content.nsdl.org/dcub/1._Intro_rev.html
-- http://content.nsdl.org/dcub/4._Elements_rev.html
-- http://content.nsdl.org/dcub/5._Qualifiers_rev.html

Diane writes: "Lots of the changes are cosmetic or editorial, and don't warrant any discussion. I took out some old stuff, like references to our abandoned vocabulary registration processes, and added to the already-present notations on Audience as the 16th element, to Audience, Provenance and RightsHolder as the additional elements not part of the original 15 of Simple DC.

"I also revised the piece of RightsHolder guidance that we discussed on the call, and I'd like some eyeballs to review. I also added a bit on the Abstract Model in section 2, which I think ought to be looked at.

"In general, I think it would be great if folks could try and look at the document with new-ish eyes, so we can discuss whether this is still a good "entry point" for users. I'm particularly concerned that we have been building up a large amount of technical rich technical documentation, which is referenced in Using Dublin Core, but not necessarily integrated very well. Is this still okay? Is there a need for some middle ground--more technical than what we have already, but still not off-putting for the non-techie? Is that possible?

"Another question is the relationship of this documentation to the newly emerging DLF OAI Best Practices documentation (see: http: //oai-best.comm.nsdl.org/cgi-bin/wiki.pl?TableOfContents) which I've been involved with, which includes a lot of references to DC, as well as other formats. Do we want to link to that, include some of it in ours, ignore it, what?"

Topic: Topic: September meeting at DC2005 in Madrid
Identifier: http://www.bi.fhg.de/People/Thomas.Baker/agenda/topic-madrid/
See also: http://www.bi.fhg.de/People/Thomas.Baker/agenda/
Created: 2005-05-11
Modified: 2005-05-13 10:52, Friday
Maintainer: Tom Baker

DC2005 in Madrid
-- http: //dc2005.uc3m.es/
Sat Sep 10: Usage Board
Sun Sep 11: Usage Board
Mon Sep 12: tutorials, papers, working groups
Tue Sep 13: tutorials, papers, working groups
Wed Sep 14: tutorials, papers, working groups
Thu Sep 15: tutorials, papers, working groups
Fri Sep 16: Advisory Board

Preliminary agenda

1. Review of Collection Description application profile

2. Revision of DCSV documents [Andy and Andrew]

3. Non-DCMI encoding schemes
Endorsement mechanism for non-DCMI encoding schemes [Stuart, Diane, Tom, Pete for RDF]
Statement of DCMI approach to endorsing vocabulary encoding schemes [Tom]

4. dc:date issues [currently with DCMI Date Working Group]
-- Encoding scheme for ISO8601
-- Possible changes to comment or definition re: ranges
-- Implications for ISO and NISO of changes to dc:date definition or comment
-- Eric Childress to report
-- note: John Kunze's RFC in pipeline

5. Documentation and maintenance [Tom]
Flagging old encoding scheme Web forms pages as obsolete
Attributes for describing DCMI terms ("status", etc)
Template posting for announcing comment period

6. Encoding scheme types - 1 hour? - unless covered already in Washington [Andy]
http: //www.bi.fhg.de/People/Thomas.Baker/agenda/topic-encoding-scheme-types
Vocabulary ES vs. syntax ES. Consequences for documentation.

7. DCMES Definitions - 1 hour? - unless covered already in Washington [Andy]
http: //www.bi.fhg.de/People/Thomas.Baker/agenda/topic-dcmes-definitions
Wording of definitions in light of DAM.
Consequences for standards.

Topic: Definitions and Labels
Identifier: http://www.bi.fhg.de/People/Thomas.Baker/agenda/topic-definitions-labels/
See also: http://www.bi.fhg.de/People/Thomas.Baker/agenda/
Created: 2005-05-13
Modified: 2005-05-13 10:52, Friday
Maintainer: Tom Baker

1. Wording of Definitions in light of Abstract Model [Andy]

In Washington, we should decide on possible changes, as outlined in [1]. We should also consider the impact of such changes on standards.

2. Labels for Element Refinements

Pete has noted in the past that some of the labels on our original element refinements are not as clear as they might be. The current labels follow the current names (which were assigned in the 'dotted concatenation' era - sometime pre-neolithic I think!) - but there is no requirement that they do.

Here's a set of possible revisions to the current labels:

Alternative -> Alternative Title
Available -> Date Available
Created -> Date Created
Issued -> Date Issued
Modified -> Date Modified
Spatial -> Spatial Coverage
Temporal -> Temporal Coverage
Valid -> Date Valid

3. Labels for Encoding Schemes

The labels for some of our encoding schemes are also not very helpful - particularly to people outside the library domain. Here's some possible revisions:

DDC - > Dewey Decimal Classification (DDC)
IMT -> Internet Media Type
LCC - > Library of Congress Classification (LCC)
LCSH -> Library of Congress Subject Headings (LCSH)
MeSH -> Medical Subject Headings (MeSH)
TGN -> Getty Thesaurus of Geographic Names (TGN)
UDC -> Universal Decimal Classification (UDC)
URI -> Uniform Resource Identifier (URI)
W3C-DTF -> W3C Date Time Formats

4. Definitions for Encoding Schemes

Definitions for things like DDC could be improved. Currently, they simply repeat the name or the label.

5. Other labels

a. "Subject and Keywords" to "Subject or Keywords" (or even "Subject Classification or Keywords") -- on the basis that the value is one or the other.

b. "Resource Type" to "Type" (on the basis that this label is the only one to refer to the Resource explicitly - i.e. we don't use things like 'Resource Title'.

c. "Rights Management" to "Rights" (on the basis that Rights Management sounds too much like DRM, which dc:rights really

isn't about - dc:rights is about asserting rights, but
not about the M !).

d. "Resource Identifier" to "Identifier".

[1] http://www.bi.fhg.de/People/Thomas.Baker/public/2005-02-28.dcmes-definitions.txt

Topic: Encoding Scheme Types
Identifier: http://www.bi.fhg.de/People/Thomas.Baker/agenda/topic-encoding-scheme-types/
See also: http://www.bi.fhg.de/People/Thomas.Baker/agenda/
Created: 2005-05-12
Modified: 2005-05-13 10:52, Friday
Maintainer: Tom Baker

For each of the currently recommended encoding schemes we need to determine if it is a Syntax Encoding Scheme or a Vocabulary Encoding Scheme.

In the DCMI Abstract Model, these are defined as follows:

> Each resource may be a member of one or more classes. Note that where the resource is a value, the class is referred to as a vocabulary encoding scheme.
>
> Syntax encoding schemes are also known as 'datatypes' in some contexts.

In Washington, we need to decide on the following:

1) Whether we accept Andy's criteria for deciding on the type of encoding scheme and his interpretation of the existing encoding schemes in light of those criteria [1].

   [1] http://www.bi.fhg.de/People/Thomas.Baker/public/2005-05-12.encoding-scheme-types.txt

2) How a decision to declare these types should be reflected in DCMI documentation. Specifically, the "Type of Term" is currently shown in the Terms source data (which is used to generate Web pages and RDF schemas) for each encoding scheme as:

   http: //dublincore.org/usage/documents/principles/#encoding-scheme

   Do we want to change each of those into one of the following:

   http: //dublincore.org/usage/documents/principles/#vocabulary-encoding-scheme
   http: //dublincore.org/usage/documents/principles/#syntax-encoding-scheme

   In

   http: //dublincore.org/usage/xml/terms.xml

   Which is used to generate:

   http: //dublincore.org/usage/terms/history/
   http: //dublincore.org/documents/dcmi-terms/
   http: //purl.org/dc/terms/

   Note that this change would trigger the creation of a new historical version for each encoding scheme (but with no consequences for term URIs).

3) Decide whether this difference needs to be further emphasized in DCMI documentation -- e.g., by splitting the section "Encoding Schemes" in the DCMI Terms document into separate sections on "Syntax Encoding Schemes" and "Vocabulary Encoding Schemes".

Decision on "Accessibility" - summary

2005-04-13

Background

-- The proposal
   http://www.ozewai.org/DC-term-proposal/prop-reqs-table2.html
   http://dublincore.org/usage/meetings/2004/10/Meeting-packet.pdf, p.127

-- About the proposal
   http://www.ozewai.org/DC-term-proposal/overview.html
   http://www.ozewai.org/DC-term-proposal/criteria.html
   http://www.ozewai.org/DC-term-proposal/index.html

-- 2005-10-01: agenda item, Shanghai meeting
   http://dublincore.org/usage/meetings/2004/10/ISSUES/terms-accessibility/

Proposed definition

    Definition: "A description of the qualities of the resource
    in terms of control, display and content that can be used to
    match the needs and preferences of a user."

    Comment: "The qualities of control, display and content
    include the user's control of the interface, the sensory
    modality of the resource as presented and variations in the
    expressive form of the information. Recommended best practice
    is to express the value in a machine-addressable manner such
    as the W3C Evaluation and Reporting Language (EARL)."

UB decision from the meeting notes (2005-10-10)

    Agreed to change definition to: "A description of the
    qualities of the resource in terms of control, display
    and content that can be used to match the needs and
    preferences of a user."

    Note: need to clarify comment to indicate what is meant
    by 'control, display and content' and to note that recommended
    best practice is to provide a machine-readable statement.

Subsequent discussion on the proposed definition:

-- One potential source of confusion is the notion of
   "access" in "accessibility", because "access" is used
   in DCMI terms and in other contexts to refer to much
   different things.

   -- The Oxford American dictionary defines "access" with
      words such as "reached, entered, or used". Other
      dictionaries define "access" in terms of "approaching,
      entering, exiting, communicating with, or making use of"
      or "freedom or ability to obtain or make use of".

   -- In the DCMI term "accessRights", access is about
      obtaining a resource (as in "obtain a copy of").
      The proposed "Accessibility" element is not about this
      type of access.

   -- "Access" is used in the context of the Budapest Open
      Access Initiative specifically to refer to
      "free availability on the public Internet" (see
      http://www.earlham.edu/~peters/fos/boaifaq.htm#openaccess).
      In other words, it also is about access in the sense of

"obtaining", which is not what the proposed Accessibility term is about.

-- In the W3C world, accessibility is also about "intellectual accessibility" -- things like level of difficulty and language. It also seems to have something to do with the ease with which a resource can be used. According to http://www.w3.org/WAI/ut2/accessibility.html, Web accessibility means that people with disabilities can use the Web. More specifically, Web accessibility means that the Web is designed so that people with disabilities can perceive, understand, navigate, and interact with it effectively, as well as create and contribute content to the Web. Web accessibility addresses all disabilities, including visual, auditory, physical, speech, cognitive, and neurological disabilities.

-- Bottom line: access-as-obtain and access-as-use are different things, even if the word "access" is often used to express both notions. Somehow the definition of the new property needs to reflect that distinction.

-- Moreover, the proposed Accessibility element really seems to be specifically about "Web Accessibility" (i.e., accessibility to digital resources in the context of the Web), though this is not stated explicitly in the definition or comment. Could one, for example, use Accessibility to describe the accessibility of a building? As proposed the term could be called "Web Accessibility", however this would seem to be a refinement of a broader notion of "Accessibility" -- not limited to digital resources. In fact, calling the term "Accessibility" when "Web Accessibility" was meant would make it difficult at a later date to create a broader term for accessibility. The Usage Board therefore felt it made more sense to broaden the definition of Accessibility explicitly to to include non-digital resources (and recognizing that "Web Accessibility" could in principle be proposed subsequently as a refinement of "Accessibility").

A broader definition was proposed, not limited to Web resources and Web technology (2005-03-18):

> Definition: "A statement about the ease with which the resource can be accessed and used, regardless of the technology being used."
>
> Comment: "Factors that determine the ease of access and use of the resource include the control of the interface, the sensory modality of the resource as presented and variations in the expressive form of the information. Recommended best practice is to express the value in a machine-readable manner such as the W3C Evaluation and Reporting Language (EARL)."

Then questions were raised with regard to how the intended use fits with the DCMI Abstract Model:

-- A statement made using the accessibility property doesn't actually describe the "accessibility" of the resource. Rather, it provides a basis for a process to take place -- involving a description of the resource and a description of the user -- the outcome of which is an indication of the accessibility of that resource for that user.

-- In the understanding of the Usage Board, the Accessibility property is intended to describe a relation between a

resource and a _description_ of that resource -- where that description describes those specific attributes of the resource that support such an "accessibility assessment" process.

-- The only descriptions the UB could find of how the "accessibility-related attributes description" is represented refer to XML -- document-based, rather than statement-based (see http://www.imsproject.org/accessibility/accmdv1p0/imsaccmd_infov1p0.html). Given that the DCMI Abstract Model (now a recommendation) is closely aligned with RDF, it would be useful to understand how such an accessibility assessment process would be modeled in RDF. In the understanding of the UB, the information that is represented in an "accessibility-related attributes description" is a set of statements about the resource -- the same resource as in the first description -- and in an RDF implementation there would be a set of properties to represent this (about use of colour, use of audio etc). In an RDF environment, it is not clear why one would actually need an "accessibility" property at all, as the set of statements could be stored as a physical RDF/XML doc separate from the resource discovery description and one could use rdfs:seeAlso to point to "more statements" about the subject resource.

-- Such a use is not without precedent, as that is how some implementers understand dc:rights. However, dc:rights may contain a string value made up of a rights statement. By analogy, it should be explicitly permissible to use an Accessibility element for an unstructured plain-literal "accessibility statement" (i.e., in words, as a string value).

-- Changing "machine-addressable" to "machine-readable" does not really address the underlying issue because plain string value and value URIs are all "machine-readable". Rather, the real issue seems to be its intended use for citing related descriptions. If this is the case, then it is perhaps more helpful to say this in the comment, pointing users to the Abstract Model.

-- It was noted that defining dcterms:accessibility as a term which links to a separate description of the resource seems a bit like creating a new term called dcterms:libraryOrientedDescription and then using it to link to a MARC record.

-- As a minor point, we agreed that "sensory modality" seems like specialist jargon and needed to be re-worded for understandability.

Discussion of these modeling issues led us to take a closer look at EARL itself:

-- The "Example" entry in the table in [1] refers to IMS AccMD XML, and the "Comment" referred to [2,3], but neither illustrates exactly how it should be implemented.

[1] http://www.ozewai.org/DC-term-proposal/prop-reqs-table2.html
[2] http://www.imsproject.org/accessibility/accmdv1p0/imsaccmd_oviewv1p0.html
[3] http://www.imsproject.org/accessibility/accmdv1p0/imsaccmd_bestv1p0.html

-- The EARL 1.0 working draft spec [1] didn't seem to be linked from the EARL home page [2]. EARL development now seems to be under [3]. The status of these various documents is was not clear to us.

[1] http://www.w3.org/TR/EARL10/
[2] http://www.w3.org/2001/03/earl/
[3] http://www.w3.org/WAI/ER/

-- In section 3 of EARL 1.0, there is an example of an EARL assertion [1]. The (newer?) EARL 1.0 example is modelled slightly differently from the (older?) example shown on the EARL home page. That old example seemed to be a simple "resource:X earl:passes test:Y" (with reification to say who asserted that). The (newer?) model sidesteps reification and has the "assertor" (and also the subject resource, the test, the result etc) as a property of a resource of type earl:Assertion. This new EARL 1.0 model would seem to fit less cleanly into the DC "statements-about-the-subject-resource" framework (there is no longer an earl:passes property involved), so on this basis, there may be more grounds for that pointer-to-EARL-stuff property.

[1] http://www.w3.org/TR/EARL10/#assertion

-- If EARL is about (to quote the homepage): "evaluating web pages and web sites against the Web Accessibility Content Guidelines", then it is arguable whether the definition should be "A statement about the ease with which it is possible ...". Rather, it should be "A statement about the conformance of a resource to a particular accessibility benchmark" or words to that effect.

-- As implied in the proposed wording of the definition of Accessibility, the intended value would seem to be a second (albeit fairly long and complex) description of the resource -- one that uses attributes of the resource that are relevant in the context of how accessibile it is likely to be - color, use of images, etc.

-- The problems we had in finding good examples as a guide served to underline a more general, medium-term need for user guidance on the intended use of an Accessibility property. In the short term, to help the Usage Board finalize this decision, the Usage Board would like to see a simple concrete example of how a DC description and an EARL assertion are used together and exactly where the proposed property fits in. More to the point, the current wording of the Comment (below) still refers to EARL, but it is unclear whether it is yet appropriate to do so, and if so, what exactly should be cited.

This discussion has led the Usage Board to the current wording on the table:

> Definition: A statement about the ease with which it is possible to use, interact with or comprehend the resource, regardless of the technology used.
>
> Comment: Ease of access encompasses the ability to perceive, understand, navigate, or interact with a resource -- digital or physical -- regardless of the technology being used. For example, this element may be used for statements about the accessibility of a resource to people who are permanently or temporarily disabled. Users wishing to associate this element with a description of a resource from the standpoint of accessibility -- a "related description" in terms of the DCMI Abstract Model [CITE] -- should consider expressing the value in a machine-readable language such as the W3C Evaluation and Reporting Language (EARL) [CITE].

Selected postings, Feb 17 to Mar 23 discussion of Accessibility

---

Date: Thu, 17 Feb 2005 08:10:47 +0000
From: Pete Johnston <p.johnston@UKOLN.AC.UK>
http://www.jiscmail.ac.uk/cgi-bin/webadmin?A2=ind0502&L=dc-usage&T=0&O=A&P=4035

Quoting Andy Powell <a.powell@ukoln.ac.uk>:

> Well, I guess I am guilty of having missed the subtle point about matching
> two sets of metadata - though I think this needs to be brought out in the
> comment rather than in the definition.

Not sure these thoughts are helpful at this point! ;-)

First, I also had some doubts about the name of the property. I had grasped this aspect of a process matching resource description and user description, and for this very reason I always wondered about whether "accessibility" was the right name for a property of a resource ;-) Because (according to this very principle of accessibility being the result of a process), a statement made using the accessibility property doesn't actually describe the "accessibility" of the resource, if you see what I mean! It just provides a basis for a process to take place (involving a description of the resource and a description of the user), the outcome of which is an indication of the accessibility of that resource for that user.

Secondly, I must admit I was struggling to grasp the underlying model here. I did have a couple of exchanges with Liddy and one of her colleagues before Shanghai, but they weren't able to provide any examples of how this was implemented in RDF (and I got too busy and didn't pursue it). It seems to me that as proposed the accessibility property describes a relation between a resource and a _description_ of that resource - where that description describes those specific attributes of the resource that support the "accessibility assessment" process). The information that is represented in this "accessibility-related attributes description" is a set of statements about the resource - the same resource as in the first description - and in an RDF implementation there would be a set of properties to represent this (about use of colour, use of audio etc. It seems to me in this model you would never actually need an "accessibility" property at all! If this set of statements was stored as a separate physical RDF/XML doc from the resource discovery description, then you'd just use rdfs:seeAlso to indicate there was more stuff about the subject resource.

But the only descriptions I can find of how the "accessibility-related attributes description" is represented refer to XML - document-based, rather than statement-based, specifically

http://www.imsproject.org/accessibility/accmdv1p0/imsaccmd_infov1p0.html

so it doesn't really help disentangle this.

Having said all this, I'm conscious that the dc:rights property takes a very similar approach to that suggested for the accessibility property - I think effectively often a dc:rights property points to another "description" (which has statements about the same resource covering those attributes of the resource that are concerned with rights) - so there is a precedent. OTOH, I think this aspect of dc:rights has caused us a few headaches as well, and I'm not sure that if

we were starting again we'd take the same approach.

Oh dear. Not sure that was constructive at this point in
the proceedings!

Date: Thu, 17 Feb 2005 18:49:37 -0000
From: Pete Johnston <p.johnston@UKOLN.AC.UK>
http://www.jiscmail.ac.uk/cgi-bin/webadmin?A2=ind0502&L=dc-usage&T=0&O=A&P=3829

Hi Stuart,

> Pete, I believe that your description of how some see the
> property used and and your comparison with rights is correct.

OK, thanks.

> However, as comments from others on the UB indicate (see
> Diane's previous post) indicate, that is not the _only_ way
> the property may be used and, in fact, any hint of of that
> specific use was moved from the definition to the comment (as
> best practice) in order to generalize the semantics of the
> property. So, just as rights may contain a string value made
> up of a rights statement, but can also reference an separate
> rights description, so may the accessibility property.

OK.

> I'd like to hear more regarding your statement: "I think this
> aspect of dc:rights has caused us a few headaches as well,
> and I'm not sure that if we were starting again we'd take the
> same approach."

I think my concern touches on your comment here. dc:rights and the
proposed property (maybe dc:description too) adopt a slightly different
modelling "style" from the other properties. What I mean by that is I
think most of the other properties are deployed to describe a
relationship between the subject resource and another thing

resource-R is-created-by person-P
resource-R has-as-topic concept-C

and so on.

OK, so does dc:rights, and the other thing is a "rights statement". OK,
but suppose I have a description which includes:

resource-R has-rights-statement statement-S

Now, in the case of dc:rights that other thing is, or a least may be, or
may include, a description of, resource-R! i.e. statement S might be an
RDF/XML document containing the statement

resource-R is-available-under-license license-L

So, hang on a minute... why didn't we just say that (using
dcterms:license)? What is the difference between the two approaches?

And why do we say

resource-R has-rights-statement statement-S

rather than

resource-R see-Also statement-S

(using rdfs:seeAlso, which is how you would usually say "more data about

this over here").

As you point out, I think the reason is that we are trying to allow for
both unstructured plain literal "rights statements" and resources as
rights-statements.

I think this really goes back to the more general value-as-resource v
value-as-literal debate. Most RDF vocabularies (I think) don't allow
this choice. Properties which can have literal values are defined to
always have literal values; properties that have other resources as
values don't take literals as values. Now in the Abstract Model, we've
said values are always resources and although we haven't worked out how
this is going to be implemented in RDF, I'm guessing we'll say the
dc:rights RDF property never has a plain literal value, but it could
have some intermediate node with a literal value hanging off it. I'm not
sure that really solves the problem but I haven't thought that bit
through....

I dunno... maybe it's all a non-problem, but the fact that it leads to
some slightly odd questions leaves me with niggling doubts about whether
the underlying approach is quite right. But I appreciate that is a very
vague specification of the problem! ;-)

I was/am really curious to see how the Accessibility people address this
in RDF partly because, well, I just want to know (!), and partly because
I think if they have worked through some of these issues it would
provide some very useful models to guide us. (Andy and I had a brief
chat about this earlier and he commented that similar issues will arise
in the ODRL-DC work that is in progress.) But most of the references I
find are to EARL, and all that material seems to be dated 2001 and I'm
not clear what its status is. And while I think I understand broadly
what EARL does, I'm rather less clear about how it integrates with e.g.
a DC metadata description. From my fairly hasty reading of it, I'm not
really sure I see a need for a dcterms:accessibility property to
reference an EARL statement, just for rdfs:seeAlso to say "there's some
more RDF data over here".

Date: Thu, 31 Mar 2005 11:14:26 +0100
From: Andy Powell <a.powell@UKOLN.AC.UK>

On Thu, 31 Mar 2005, Thomas Baker wrote:

> -- "Access" is used in other DCMI definitions to mean
> something different. In particular, "accessRights" is
> about who can "access" a resource more in the sense of
> "reach and use".

Hmmm... actually, I disagree with this. Access Rights is
about whether someone has rights to obtain (a copy of) the
resource. It is not primarily about 'use' I don't think? So,
for example, access to a bit of shareware software may be free
provided that a registration process is gone thru but the usage
of that software may be limited by licence to 1 month's use.
Access Rights is about the former but not about the latter in
my view. In DCMI terms, the latter is now covered by License.

As an aside, given your dislike of defining Accessibility using the word
'access', I note that in the definition of Access Rights we do not really
define what we mean by access (which is used in both the definition and
the comment). Oh well :-(

> Definition: A statement about the ease with which
> the resource can be obtained or used.

Sorry, but my personal view is that replacing 'accessed' by 'obtained'
here really doesn't help much. In fact, the combination of 'obtained' and

'or' is actually worse, since it allows a statement to be made only about how easy it is to obtain the resource - which IMHO is not necessarily an 'accessibility' statement at all.

In any case, I thought you wanted to differentiate the 'access' in accessibility from the 'access' in access rights. By using 'obtain and use' you made them exactly the same ('reach and use')?? If access means the same in both cases, why not just use 'access' in both?

So, I would prefer to stick with 'accessed and/or used' with 'obtained and/or used' being second best.

I quite like your comment, though I note that having removed 'accessed' from the definition, it feels odd to me to start with 'Ease of access encompasses ...'??

> Comment: Ease of access encompasses the ability to
> perceive, understand, navigate, or interact with a
> resource -- digital or physical -- regardless of the
> technology being used. For example, this element may
> be used for statements about the accessibility of a
> resource to people who are permanently or temporarily
> disabled. Users wishing to associate this element
> with a description of a resource from the standpoint
> of accessibility -- a "related description" in terms
> of the DCMI Abstract Model [CITE] -- should consider
> expressing the value in a machine-readable language
> such as the W3C Evaluation and Reporting Language
> (EARL) [CITE].

The problem I think is that with both your definition and mine, the statement "The resource is freely available" is a valid value (given the definitions and without looking at the comments) but isn't what we mean by accessibility.

A resource can be 'open access' but inaccessible.

I think we need to spell out what 'access' means in the context of accessibility in the definition, rather than in the comment. So, re-casting your comment slightly gives...

"A statement about the ease with which it is possible to access, use, interact with or comprehend the resource"?

And... for the record, I still think that this, and the first sentence of your comment, is just as much a definition of 'usability' as a definition of 'accessibility'. It is the 'regardless of the technology being used' bit that is critical here IMHO, and therefore that bit should remain in the definition.

Which I think leaves my preferred option as something like

"A statement about the ease with which it is possible to access, use, interact with or comprehend the resource, regardless of the technology being used"

Sorry :-(

Date: Thu, 31 Mar 2005 12:04:40 +0100
From: Pete Johnston <p.johnston@UKOLN.AC.UK>

I was keeping out of these discussions, but I skimmed Tom's proposed definition as it joined my email backlog and my reaction was similar to Andy's.

I particularly agree with this point:

> A resource can be 'open access' but inaccessible.

I don't think "accessibility" (as intended in the initial property proposal) was about accessing-as-obtaining a resource, "getting to" the resource, if you like. The initial definition proposed in

http://www.ozewai.org/DC-term-proposal/prop-reqs-table2.html

was:

====
A reference to a machine-readable profile that describes the qualities of a resource that can be used to match the needs and preferences of a user as expressed in a machine-readable user profile.
====

The proposal referenced the IMS Accessibility work, and it may be useful to look at section 3 of

http://www.imsproject.org/accessibility/accmdv1p0/imsaccmd_oviewv1p0.html

for an idea of what "qualities of the resource" were in mind:

===
Access Modality: Whether the user requires vision, hearing, touch and/or text literacy to access the resource.
Adaptability: How amenable the resource is to transformation of the display and whether the method of control is flexible (display transformability and control flexibility).
Equivalent: Whether there is a known equivalent alternative.
===

It seems fairly clearly about accessing-as-using the resource, "getting into" the resource, once I have "obtained" it - though of course the description of those attributes that condition access-as-use may be used to discover the resource in the first place.

I think access-as-obtain and access-as-use are different things, even if the word "access" is often used to express both notions, and - somehow! ;-) - the definition of the new property needs to reflect that distinction.

Date: Thu, 31 Mar 2005 16:13:39 +0200
From: Thomas Baker <thomas.baker@bi.fhg.de>

On Thu, Mar 31, 2005 at 11:14:26AM +0100, Andy Powell wrote:
> >-- "Access" is used in other DCMI definitions to mean
> > something different. In particular, "accessRights" is
> > about who can "access" a resource more in the sense of
> > "reach and use".
>
> Hmmm... actually, I disagree with this. Access Rights is about whether
> someone has rights to obtain (a copy of) the resource. It is not
> primarily about 'use' I don't think? ...

Okay, I agree; point taken.

> As an aside, given your dislike of defining Accessibility using the word
> 'access', I note that in the definition of Access Rights we do not really
> define what we mean by access (which is used in both the definition and
> the comment). Oh well :-(

Right... -- and I figured that using it in two different ways in two different definitions without defining it even once

might be going a bit too far... ;-)

> > Definition: A statement about the ease with which
> > the resource can be obtained or used.
>
> Sorry, but my personal view is that replacing 'accessed' by 'obtained'
> here really doesn't help much. In fact, the combination of 'obtained' and
> 'or' is actually worse, since it allows a statement to be made only about
> how easy it is to obtain the resource - which IMHO is not necessarily an
> 'accessibility' statement at all.

Hmm, good point.

> In any case, I thought you wanted to differentiate the 'access' in
> accessibility from the 'access' in access rights. By using 'obtain and
> use' you made them exactly the same ('reach and use')?? If access means
> the same in both cases, why not just use 'access' in both?
>
> So, I would prefer to stick with 'accessed and/or used' with 'obtained
> and/or used' being second best.

Hmm, is Accessibility really about "obtaining" at all...?

> A resource can be 'open access' but inaccessible.
>
> I think we need to spell out what 'access' means in the context of
> accessibility in the definition, rather than in the comment. So,
> re-casting your comment slightly gives...
>
> "A statement about the ease with which it is possible to access, use,
> interact with or comprehend the resource"?

Okay, I like this. I'm just wondering whether the word "access" is
really necessary...

> And... for the record, I still think that this, and the first sentence of
> your comment, is just as much a definition of 'usability' as a definition
> of 'accessibility'. It is the 'regardless of the technology being used'
> bit that is critical here IMHO, and therefore that bit should remain in
> the definition.

Okay, but is there really any reason why the element could not
or should not be called "usability" (except for the obvious
fact that it was proposed by a community that talks about
"accessibility")? Typing "usability studies" into Google,
right near the top I see a report on "usability tests with
users with disabilities" filed under "accessibility".

> Which I think leaves my preferred option as something like
>
> "A statement about the ease with which it is possible to access, use,
> interact with or comprehend the resource, regardless of the technology
> being used"
>
> Sorry :-(

It is still not clear to me why the definition needs to refer
to "technology" because, to me, there is nothing in the phrase
"to access, use, interact with or comprehend the resource"
which implies that technology is being used at all.

So my suggestion would be to use your definition, but minus
"access" and minus "regardless...":

    A statement about the ease with which it is possible to
    use, interact with or comprehend the resource.

Date: Thu, 31 Mar 2005 16:28:00 +0200
From: Thomas Baker <thomas.baker@bi.fhg.de>

On Thu, Mar 31, 2005 at 12:04:40PM +0100, Pete Johnston wrote:
> I particularly agree with this point:
>
> > A resource can be 'open access' but inaccessible.
>
> I don't think "accessibility" (as intended in the initial property
> proposal) was about accessing-as-obtaining a resource, "getting to" the
> resource, if you like. ...

Yes, I agree with this.

> for an idea of what "qualities of the resource" were in mind:
>
> ===
> # Access Modality: Whether the user requires vision, hearing, touch
> and/or text literacy to access the resource.
> # Adaptability: How amenable the resource is to transformation of the
> display and whether the method of control is flexible (display
> transformability and control flexibility).
> # Equivalent: Whether there is a known equivalent alternative.
> ===
>
> It seems fairly clearly about accessing-as-using the resource, "getting
> into" the resource, once I have "obtained" it - though of course the
> description of those attributes that condition access-as-use may be used
> to discover the resource in the first place.

This is helpful. Nothing in there about "obtaining" in the
sense of "discovering" or "fetching". Hmm, except perhaps
with regard to equivalent alternatives.

> I think access-as-obtain and access-as-use are different things, even if
> the word "access" is often used to express both notions, and - somehow!
> ;-) - the definition of the new property needs to reflect that
> distinction.

Yes, that states the contrast nicely. Since we have already
used "access" in its "obtain" meaning for accessRights, using
it in the definition for Accessibility arguably muddies the
waters by implying that "obtaining" the resource is part of
what Accessibility is about. Again:

    A statement about the ease with which it is possible to
    use, interact with or comprehend the resource.

Date: Thu, 31 Mar 2005 16:34:00 +0100
From: Andy Powell <a.powell@UKOLN.AC.UK>
Subject: Re: "Accessibility" decision - proposed wording

I don't know about anyone else, but I would find it helpful to see some
example values for Accessibility, as intended by the proposers. Can we
ask Liddy to supply some? Were there examples in the original proposal?

We know that EARL fits in somewhere, but if I'm honest, I don't really
know what an EARL statement looks like or does! My understanding is that
EARL is a language for recording the conformance of a resource against a
particular benchmark. So, in this case, EARL would presumably be used to
make a staement about whether a resource conforms to WAI or not?

>From the EARL homepage...

--- cut ---

> EARL would initially be a means for expressing in a machine readable form
> ...
>
> * Results of evaluating web pages and web sites against the Web
>   Accessibility Content Guidelines

--- cut ---

It is therefore arguable whether the definition should be "A statement about the ease with which it is possible ..." - rather it should be "A statement about the conformance of a resource to a particular accessibility benchmark" or words to that effect.

But it is also possible that the intended 'value' of Accessibility is a complete chunk of XML that conforms to the IMS Accessibility spec. If so, then things get quite complex because the IMS spec references EARL statements itself (several times).

As hinted in the original wording of the definition of Accessibility

> A description of the qualities of the resource in terms of control, display and content that can be used to match the needs and preferences of a user.

the value looks to be a second (albeit fairly long and complex) description of the resource - but one that uses attributes (qualities!) of the resource that are relevant in the context of how accessibile it is likely to be - colour, use of images, etc.

Defining dcterms:accessibility as a term which links to a separate description of the resource is a bit like creating a new term called dcterms:libraryOrientedDescription and then using it to link to a MARC record.

As PeteJ has argued in the past, if there are useful attributes defined in the IMS Accessibility spec then (provided they are declared as RDF properties) we can use them directly in our descriptions - we don't need a new Accessibility property in order to do that. (Just as we are doing with the MARC relator terms from the library world).

So, I must admit that I'm totally confused about what we are achieving (or even trying to achieve) here.

My personal view is that it looks as though we were wrong to approve this term in Shanghai - and that it is better to admit that now than to move forward on shaky ground.

I urge us not to do anything more on this without involving Liddy and without getting some concrete example values from her.

Date: Thu, 31 Mar 2005 17:07:55 +0100
From: Pete Johnston <p.johnston@UKOLN.AC.UK>

Andy said:

> I don't know about anyone else, but I would find it helpful
> to see some example values for Accessibility, as intended by
> the proposers. Can we ask Liddy to supply some? Were there
> examples in the original proposal?

The "Example" entry in the table in

http://www.ozewai.org/DC-term-proposal/prop-reqs-table2.html

refers to IMS AccMD XML, and the "Comment" referred to

http://www.imsproject.org/accessibility/accmdv1p0/imsaccmd_oviewv1p0.html
http://www.imsproject.org/accessibility/accmdv1p0/imsaccmd_bestv1p0.html

But neither illustrates exactly how it should be implemented.

> We know that EARL fits in somewhere, but if I'm honest, I
> don't really know what an EARL statement looks like or does!
> My understanding is that EARL is a language for recording the
> conformance of a resource against a particular benchmark.
> So, in this case, EARL would presumably be used to make a
> staement about whether a resource conforms to WAI or not?
>
> From the EARL homepage...
>
> --- cut ---
> EARL would initially be a means for expressing in a machine
> readable form
> ...
>
> * Results of evaluating web pages and web sites against the Web
> Accessibility Content Guidelines
> --- cut ---
>
> It is therefore arguable whether the definition should be "A
> statement about the ease with which it is possible ..." -
> rather it should be "A statement about the conformance of a
> resource to a particular accessibility benchmark" or words to
> that effect.

The EARL 1.0 working draft spec

http://www.w3.org/TR/EARL10/

didn't seem to be linked from the EARL home page

http://www.w3.org/2001/03/earl/

:-(

FWIW, EARL development now seems to be under

http://www.w3.org/WAI/ER/

Anyway, down in section 3 of EARL 1.0, there is an example of an EARL
assertion

http://www.w3.org/TR/EARL10/#assertion

(more below)

> But it is also possible that the intended 'value' of
> Accessibility is a complete chunk of XML that conforms to the
> IMS Accessibility spec. If so, then things get quite complex
> because the IMS spec references EARL statements itself
> (several times).
>
> As hinted in the original wording of the definition of Accessibility
>
> A description of the qualities of the resource in terms of control,
> display and content that can be used to match the needs and
> preferences
> of a user.
>
> the value looks to be a second (albeit fairly long and
> complex) description of the resource - but one that uses
> attributes (qualities!) of the resource that are relevant in
> the context of how accessibile it is likely to be - colour,

> use of images, etc.
>
> Defining dcterms:accessibility as a term which links to a
> separate description of the resource is a bit like creating a
> new term called dcterms:libraryOrientedDescription and then
> using it to link to a MARC record.
>
> As PeteJ has argued in the past, if there are useful
> attributes defined in the IMS Accessibility spec then
> (provided they are declared as RDF
> properties) we can use them directly in our descriptions - we
> don't need a new Accessibility property in order to do that.
> (Just as we are doing with the MARC relator terms from the
> library world).

I noticed that that (newer?) EARL 1.0 example is modelled slightly differently from the (older?) example shown on the EARL home page.

That old example seemed to be a simple

resource:X earl:passes test:Y

(with reification to say who asserted that)

The (newer?) model sidesteps reification (I think) and has the "assertor" (and also the subject resource, the test, the result etc) as a property of a resource of type earl:Assertion.

So I can see that this new EARL 1.0 model would fit less cleanly into the DC "statements-about-the-subject-resource" framework (there is no longer an earl:passes property involved), so on this basis, there may be more grounds for that pointer-to-EARL-stuff property.

> So, I must admit that I'm totally confused about what we are
> achieving (or even trying to achieve) here.
>
> My personal view is that it looks as though we were wrong to
> approve this term in Shanghai - and that it is better to
> admit that now than to move forward on shaky ground.
>
> I urge us not to do anything more on this without involving
> Liddy and without getting some concrete example values from her.

Yes, I agree.

The W3C documentation on this stuff does seem to be rather fragmented, and even when I manage to find stuff, I end up guessing about what is the most current.

And yes, I would just like to see a simple concrete example of how a DC description and an EARL assertion are used together and exactly where the proposed property fits in.

Date: Mon, 18 Apr 2005 13:58:23 +0200
From: Thomas Baker <thomas.baker@bi.fhg.de>
To: DCMI Usage Board back-channel <dc-usage-bc@jiscmail.ac.uk>
Subject: Accessibility - new proposal

Liddy has reacted to the digest of our discussions (see my summary below).

My gut feeling is that the best we can do, under the circumstances, is to finalize something rather close to what we "approved" in Shanghai. I propose:

> Definition: A description of the resource in terms of how it can be perceived, understood, or interacted with by users.
>
> Comment: An accessibility description might be used to match the (digital or physical) resource to a user on the basis of the user's ability profile. Implementers wishing to express the values for this element in a machine-readable language should consider treating such expressions as "related descriptions" in terms of the DCMI Abstract Model.

In a nutshell:
-- The definition returns to "description" (instead of "statement") -- but dc:description itself can have a literal value.
-- It is not limited to digital resources (and Liddy agrees; see below).
-- It does not cite EARL, which is not yet stable (and Charles agrees; see below).
-- It includes a caveat with regard to associating the element with a "related description".
-- It refers to the intended use ("matching"). As we did in fact "approve" such language in Shanghai, it would be awkward to argue that we have since changed our mind...

How to proceed? Liddy and Charles have not yet seen the proposed text -- I'm posting it here first. In order to post my summary of their email (below) on the public list, I'd need their approval and, ideally, confirmation that my summary is correct. Hence I'm posting first to DC-USAGE-BC...

Tom

Main background, repeated here as reference (see earlier summary of the complete discussion)

> Proposed definition
>
> > Definition: "A description of the qualities of the resource in terms of control, display and content that can be used to match the needs and preferences of a user."
> >
> > Comment: "The qualities of control, display and content include the user's control of the interface, the sensory modality of the resource as presented and variations in the expressive form of the information. Recommended best practice is to express the value in a machine-addressable manner such as the W3C Evaluation and Reporting Language (EARL)."

UB decision from the meeting notes (2005-10-10)

> Agreed to change definition to: "A description of the qualities of the resource in terms of control, display and content that can be used to match the needs and preferences of a user."
>
> Note: need to clarify comment to indicate what is meant by 'control, display and content' and to note that recommended best practice is to provide a machine-readable statement.

Definition on the table as of 11 April

> Definition: A statement about the ease with which it is possible to use, interact with or comprehend the resource, regardless of the technology used.
>
> Comment: Ease of access encompasses the ability to perceive, understand, navigate, or interact with a resource -- digital or physical -- regardless of the technology being used. For example, this element may be used for statements about the accessibility of a resource to people who are permanently or temporarily disabled. Users wishing to associate this element with a description of a resource from the standpoint of accessibility -- a "related description" in terms of the DCMI Abstract Model [CITE] -- should consider expressing the value in a machine-readable language such as the W3C Evaluation and Reporting Language (EARL) [CITE].

Liddy's reaction to the discussion and proposed definition (disclaimer: paraphrased by me).

-- The April 11 definition is not acceptable. The definition and comment make a statement that should not be made by DCMI "for many reasons". DCMI would, in effect, be "making all the mistakes of non-experts in the field".

-- For Liddy, it is important to hold on to the notion of "can be used to match the needs and preferences of a user." She suggests: "A description of the qualities of the object in terms of accessibility characteristics that can be used to match the needs and preferences of a user."

-- Charles agrees, saying that the important thing is that the metadata describes characteristics of the resource that are relevant to its accessibility, "specifically so that appropriate matches can be made between users and resources".

-- Liddy agrees about not enumerating "control, display, and content" as "qualities of a resource". She prefers a vaguer wording such as "in terms of accessibility characteristics" because this could be broadened to include physical access and whatever else might be discovered as necessary for wheel-chairs and the like. She would not want to restrict the element at this stage, so that they will be able to describe best practice in a variety of circumstances more easily.

-- She does agree with broadening the concept by removing references to specific qualities that restrict the scope of the element. That said, for the purposes of clarification:

   -- "control" is the human-computer interaction relationship;

   -- "display or presentation" has to do with the modality --

visual, auditory or tactile;

-- "content" has to do with the intellectual contribution of the resource, service, or other object.

-- Liddy agrees that access-as-obtain and access-as-use are different things. The problem with the word 'accessibility' is well-known and confusion has occurred in many fora, but it is felt that changing the term used would cause even more confusion. Best practice is to avoid the use of the word 'access' and to always structure sentences so that 'accessibility' can be used. Conversely, the accessibility community carefully avoid referring to obtainability as 'accessibility'.

-- Liddy agrees that in the W3C world, accessibility is about "intellectual accessibility" -- things like level of difficulty and language. It also seems to have something to do with the ease with which a resource can be used. According to http://www.w3.org/WAI/ut2/accessibility.html, Web accessibility means that people with disabilities can use the Web. More specifically, Web accessibility means that the Web is designed so that people with disabilities can perceive, understand, navigate, and interact with it effectively, as well as create and contribute content to the Web. Web accessibility addresses all disabilities, including visual, auditory, physical, speech, cognitive, and neurological disabilities.

-- Liddy agrees that the proposed Accessibility element is not specifically about "Web Accessibility" in the sense of accessibility to digital resources in the context of the Web. Rather, the "Web" part is the idea that the Web is used to ship around the metadata, and the resources described by that metadata need not be digital. For example, in the educational context they could be physical objects such as text books and work sheets.

-- Therefore, she agrees that one could use this element to describe the accessibility of a building. In fact, this has been considered and there is already work underway to establish a vocabulary for event and location physical accessibility (arising from last year's work in MMI-DC in Europe), though in such contexts, she thinks the definition should perhaps refer to 'object' instead of 'resources' because people don't think of buildings as 'resources'.

-- She particularly dislikes one of the alternatives we considered ("A statement about the ease with which the resource can be accessed and used, regardless of the technology being used."). The main objections:

-- it is quite different from what was proposed;
-- it does not convey the purpose of the metadata;
-- it uses the bad word (access) in a confusing way, hinting at definitions that have been discredited in the field.

-- Commenting on text we considered at one point

> Factors that determine the ease of access and use of the resource include the control of the interface, the sensory modality of the resource as presented and variations in the expressive form of the information.

...she made the following objections:

-- the word 'access' is used incorrectly

   -- there is an opening for usability in general, which
      was never intended;
   -- the term involves a quality judgment, which seems
      inappropriate for DCMI.

-- Liddy did not at all understand our confusion about the
   the Accessibility property not actually describing the
   "accessibility" of the resource, but as describing the
   resource (itself) in the context of a process -- involving
   a description of the resource and a description of the
   user -- the outcome of which is an indication of the
   accessibility of that resource for that user. Liddy says
   that an accessibility statement describes in strict detail
   the accessibility qualities of the object just like date
   has details about a date.

   In other words, she does not at all understand the notion of
   Accessibility as denoting a relation between a resource and
   a _description_ of that resource -- where that description
   describes those specific attributes of the resource that
   support such an "accessibility assessment" process.

-- Liddy agrees that we should be able to use an Accessibility
   element for an unstructured plain-literal "accessibility
   statement". The example she offers is: "This resource
   conforms to WCAG Level AA".

-- Confusingly -- in light of her suggestion above! -- Liddy
   strongly objects to defining Accessibility as "A statement
   about the conformance of a resource to a particular
   accessibility benchmark" (or words to that effect) --
   a wording we considered because it seemed to be what
   some of the Web pages were about. She explains that the
   accessibility community has done alot of work to get away
   from this approach. Conformance statements in this field are
   very problematic and not helpful anyway as there is no known
   specification to which conformance guarantees accessibility.

-- With regard to our confusion about EARL, she reports that
   David Weinkauf and Anastasia Cheetham have developed a small
   EARL generator that produces what they are talking about --
   see http://tile-gus.atrc.utoronto.ca/acheck/index.html.
   She points to http://inclusivelearning.ca/TILE/ for a
   working demo of a production system using this approach.

-- Following URLs she provided, I found
   http://www.imsproject.org/accessibility/accmdv1p0/imsaccmd_bestv1p0.html
   -- a user guide which explains EARL on the example of TILE.
   This makes the intended process a bit clearer to me.
   Basically, it involves comparing "profiles" in XML --
   on one hand the profile encoding the accessibility needs
   and preferences of a user, on the other a profile associated
   with a resource (which is what they want to point with using
   the Accessibility element):

        When the TILE authoring tool is used to aggregate
        and publish learning objects, authors are prompted
        to provide information about the modality of the
        resources, stating whether or not they contain
        auditory, visual, textual, or tactile content, as
        well as any equivalent alternative resources along
        with their alternative accessibility properties. This
        information is captured in the resource meta-data
        profile. Users, on the other hand, are given the
        option of creating an ACCLIP profile stating their
        accessibility needs and preferences. Together, this
        information is used to determine whether or not a
        requested primary resource should be substituted or

supplemented with an equivalent alternative resource, as well as styled (e.g., CSS) or transformed (e.g., image to ALT text), in order to meet the needs and preferences of the user.

-- Charles thinks the technology is reasonably stable in practice but agrees that EARL is not "stable" in terms of process. He therefore agrees that we might drop the reference to EARL, though thinks it is important to recommend something machine-readable. Since DC allows literal text, he figures a plain-text accessibility statement would would be feasible though not very useful.

Date: Thu, 21 Apr 2005 14:20:34 -0400
From: "Weibel,Stu" <weibel@OCLC.ORG>
Subject: Re: Definition
Comments: cc: Liddy Nevile <liddy@sunriseresearch.org>
To: DC-USAGE@JISCMAIL.AC.UK

I've had a private conversation with Tom about the definitions below, and with Tom's consent, I've offered to share my own observations. These remarks are public, open, and shareable with others, and of course brickbats are always welcome. I've corrected in the text the typos Tom told me about.

1. A description of the characteristics of a resource that may be relevant to the accessibility needs and preferences of a user.

2. A description of the characteristics of the resource which may enable or inhibit users' sensory perception or control of the resource or their intended engagement with the resource given their special needs in the context.

In the interest of full disclosure, I'll say that Liddy shared with me an earlier version, and I made some recommendations that seem to have migrated (at least partially) into the first. The first one is not too bad in my estimate, though there are a couple words I'd still want to eliminate, but it is readable and, with the exception of the circularity of using the word 'accessibility' in the definition, understandable.

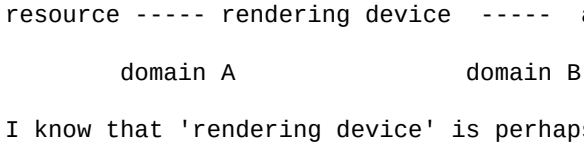
The second definition I find overwrought. It is twice as long and half as clear. No one will know what it means. Approval of such a definition would, in my estimation, elevate confusion to new heights.

I am given to understand that part of the problem of wording is to make the definition sufficiently abstract that things beyond information resources can be described with it (buildings, events, and such). This is probably a desireable goal... we've always tried to make DC as general as we can. But it substantially complicates the definition to do so. I don't have an immediate solution for this (alert the media) but I would like to give it a try, perhaps in a subsequent post.

I do have a proposal about the underlying model for an accessibility term that might simplify the task. I know I have not been party to these discussions, and if my remarks are naïve or just wrong, ok, but I believe that you should not reject my model without clear examples of how it fails.

My model:

Accessibility issues take place in a tripartite context:



I know that 'rendering device' is perhaps too info-asset-specific... I don't have a better term at the moment, but I'm confident that can be solved.

My assertion is that the definition can be made far more succinct and clear if one avoids issues of 'user preference' and 'user capabilities' (domain B interactions). Accessibility metadata should not be constructed to address those issues,

or cognitive issues, but rather to make it possible the
matching of device capabilities and resource attributes
(domain A interactions).

Domain B is important, of course, but is accommodated by a
marketplace that produces devices (screen readers, magnifying
glasses, wheel chairs, canes, crutches, etc).

The question for accessibility metadata is simply to
make it possible for a system to identify appropriate
resources that can be used in a given context.
(building_has_wheelchair_ramps, Device_renders_HTML_to_voice,
resource_encoded_according_to_ISO_12345, etc).

So now for the brickbats....?

stu

Date: Thu, 21 Apr 2005 14:38:42 -0400
From: "Weibel,Stu" <weibel@OCLC.ORG>
Subject: A variant proposal for an accessibility definition
Comments: cc: Liddy Nevile <liddy@sunriseresearch.org>
To: DC-USAGE@JISCMAIL.AC.UK

My model from my previous post:

Accessibility issues take place in a tripartite context:

resource ----- rendering device ----- agent

domain A domain B

Domain A being the technical matching domain, and domain B being the
social and economic matching of devices between people and devices.

In thinking about this, it occurs to me that the marketplace rarely, if
ever, builds rendering devices without a clear specification to build
to, nor are buildings built to accommodate arbitrary access issues, but
rather are built to one or another legal specification.

Why not structure accessibility metadata to reflect this? Its possible
that some cases might be missed, but isn't it the case that what we are
trying to promote is a declaration of conformance? This averts the
problem of specifying a user's need/disability, and also the problem of
finding a generalization for the 'rendering device'.

The definition might then become:

1. A assertion of conformance of a resource to a public standard,
specification, or legal requirement pertaining to accessibility.

The acceptable values for such an element might include:
- a formally maintained URI pointing to the standard, specification,
or legal requirement (preferred).
- the Title of the standard, specification, or legal requirement
(less desireable, but probably still useful).

Date: Thu, 21 Apr 2005 16:23:56 -0400
From: "Weibel,Stu" <weibel@OCLC.ORG>
Subject: Re: A variant proposal for an accessibility definition
Comments: cc: Liddy Nevile <liddy@sunriseresearch.org>
To: DC-USAGE@JISCMAIL.AC.UK

Hi, Pete,

Thanks for quickly responding to my modest proposal.

I see the logic in your argument... It is homologous, at least to a degree, to the educational standards element.

I personally have no considered opinion on whether this is in any important way different.

The essence of my proposal is, I believe, agreeing on the model and promoting practice based on that. I am very curious as to whether the model is acceptable to the accessibility group. Indeed, my own understanding of the problem in this way originated with discussions I had with Liddy on the topic perhaps 2 years ago. So, I was surprised to find that it is not currently the way they are thinking.

stu

-----Original Message-----
From: A mailing list for the Dublin Core Metadata Initiative's Usage Board [mailto:DC-USAGE@JISCMAIL.AC.UK] On Behalf Of Pete Johnston
Sent: Thursday, April 21, 2005 4:12 PM
To: DC-USAGE@JISCMAIL.AC.UK
Subject: Re: A variant proposal for an accessibility definition

Hi Stu,

> The definition might then become:
>
> 1. A assertion of conformance of a resource to a public standard,
> specification, or legal requirement pertaining to accessibility.
>
> The acceptable values for such an element might include:
> - a formally maintained URI pointing to the standard,
> specification, or legal requirement (preferred).
> - the Title of the standard, specification, or legal requirement
> (less desireable, but probably still useful).

In the terms of the Abstract Model, that URI and that title would be a "value URI" and a "value string" respectively, and the "value" would be the standard/specification/legal requirement itself.

However, taking the proposed definition and the comment about the values together, it is slightly ambiguous whether the proposal is that

- the value is a standard
- the value is an assertion of conformance to a standard

I think there is a slight difference between those two things.

If the value is a standard, and a statement made using the property dcterms:accessibility indicates an assertion of conformance to the standard (which I'm fairly sure is the intent of your proposal), then this seems pretty close to the existing property dcterms:conformsTo, defined as

> A reference to an established standard to which the resource conforms.

(which if it was revised to be more DCAM-compatible would drop the "A reference to" bit).

The only difference is the qualification that in the dcterms:accessibility case it is a standard "pertaining to accessibility". If this is the case, is a new, separate property necessary? If a new property is necessary, then it seems to me it would be a subproperty of dcterms:conformsTo....

To: DC-USAGE-request@jiscmail.ac.uk
From: Liddy Nevile <liddy@SUNRISERESEARCH.ORG>
Subject: Re: Accessibility definition
Date: Fri, 29 Apr 2005 22:05:11 +1000

---

what Stu says next is not at all to the issue, I am afraid.
The underlying model is not correct, sorry.

> I do have a proposal about the underlying model for an accessibility
> term that might simplify the task. I know I have not been party to
> these discussions, and if my remarks are naïve or just wrong, ok, but
> I believe that you should not reject my model without clear examples
> of how it fails.
>
> My model:
>
> Accessibility issues take place in a tripartite context:
>
> resource ----- rendering device ----- agent
>
> domain A domain B
>
> I know that 'rendering device' is perhaps too info-asset-specific... I
> don't have a better term at the moment, but I'm confident that can be
> solved.

This was an early idea about how to deal with accessibility and it does
not work.

> My assertion is that the definition can be made far more succinct and
> clear if one avoids issues of 'user preference' and 'user
> capabilities' (domain B interactions). Accessibility metadata should
> not be constructed to address those issues, or cognitive issues, but
> rather to make it possible the matching of device capabilities and
> resource attributes (domain A interactions).

Stu seems to be straying into device independence issues. Needs and
preferences are vital in the accessibility context. People with
disabilities often suffer from changes in their abilities due to such
factors as tiredness etc and they also often have rights such as that
they should have access to a dictionary that are then set aside in some
special circs - eg when they are doing a spelling test. All these
things have been worked out by people who understand the field and it
is not a good idea to just guess what people with disabilities need.
It is also not true that we don't need to think about cognitive
disabilities etc - very smart people have car accidents and strokes and
their cognitive capacity can be affected in a variety of ways - people
with dyslexia etc also are considered to be people suffering
disabilities. Such people need to be accommodated.

> Domain B is important, of course, but is accommodated by a marketplace
> that produces devices (screen readers, magnifying glasses, wheel
> chairs, canes, crutches, etc).

This is not the case.

Index of documents for DC Adaptability Proposal - updated May 2005

This proposal, initiated in October 2004, is for consideration by the Dublin Core Usage Board. The proposed element has a history that is considered a strength of the element but it does require careful reading of the following documents to recognise what has happened. **Please note** that all the following documents have been re-written for the Usage Board meeting in May 2005.

The following DC documents are available:

Overview

This document explains the proposed term and why it is being proposed as a DC term.

Criteria

This document addresses the criteria used by the Usage Board to make decisions about new terms.

Decision Tree Table

This document is self-explanatory as DC UB's decision tree

Proposal Requirements Table

This document contains the information required by the Usage Board for new terms.

Usage of the DC Adaptability term

It is anticipated that a number of communities will use the DC Adaptability term but the first is the community concerned about acessibility, particularly for people for whom special attention is required immediately. Currently, people who are blind, who cannot ever use their hands, and others are denied access to almost evey resource on the Web, including those they could use, simply because they cannot find them. Applications are beginning to use the term to identify resources that can be adapted for such people.

A typical community is those who provide learning resources for students using the IMS Global Learning Consortium (IMS) specifications. That community is seeking ways to integrate the requirements identified in the IMS/DCMI work to match digital resources to students. Others from that community, working with CEN/ISSS APLR are working on how they will also match mixed digital/physical resources to students. Yet another community group is those working on the ISO JTC1 SC 36 standard for students who anticipate working immediately with digital and then mixed learning resources, and then possibly educational events and places. The DC Adaptability term should serve all these communities. See http://www.imsglobal.org/accessibility/.

4.3.1. Criteria for evaluating a term proposal

4.3.1.1. Clarity

4.3.1.1.1. Can the term be clearly defined?

Yes. This is an element the value of which is intended to be used by computers. This means that the value of the element is best if machine readable but can be useful even if human-readable only. It could be either a URI, some encoded text, words from a controlled vocabulary or free text.

4.3.1.1.2. Can the semantics of the proposed element or element refinement be expressed precisely, unambiguously, and briefly?

Yes. The element definition is precise, unambiguous and brief.

4.3.1.2. Practicality

4.3.1.2.1. Is the term practical?

4.3.1.2.2. How difficult would it be for people creating metadata to comprehend the semantics of the proposed element or element refinement and to apply it reasonably in the description of resources?

When a user cannot hear dialogue in a reource, they may want text captions. The availability or otherwise of captions is easy to indicate. Vocabularies of such characteristics can be specified to make it easier for metadata creators (these have already been suggested as part of the work already completed). In cases where adaptability is harder to determine, such as the trasformability of text, it is usual to use accessibility tools to determine this, and the tools provide machine and human readable descriptions.

4.3.1.3. Placement

4.3.1.3.1. Does the term refine an existing element?

No, although it could be argued it combines refinements of a number of them. Early work on adaptability showed that it would not be easy to include the necessary information and that if it was included in amongst the general set of metadata terms, it would be spread all over the place and several term values would have to be brought into comparison to determine what is necessary. For instance, while the genre of a resource may be ‘text' with the implication that it is to be read, the format might be ‘image'. In this case, the resource might be inaccessible to some users. Neither value in isolation can provide this information. Knowing if the text is or is not transformable usually requires more information than just these two values that together only predict there might be transformability problems.

For a further explanation of these problems, see the report of the DC Accessibility Working Group meeting held at the DC Conference in Florence (http://dublincore.org/groups/access/workshop-20021017.html).

4.3.1.3.2. If the proposed term is an element, can it reasonably be handled as effectively as an element refinement or encoding scheme for an existing element?

No. (Attach notes about the existing DC terms and why they do not solve the problem – see 4.3.1.3.1.)

4.3.1.3.3. Are there alternative ways of implementing the term? Within the conceptual framework of the Dublin Core Element Set (i.e., element/element refinements and encoding schemes), are there alternative ways to achieve the ends sought?

It would be possible to work some of the aspects of the adaptability agenda into a number of parts of the existing DC structure. There is an explanation of the technical problems with this. More importantly, the value of a separate, self-contained element has been considered in detail. Generally, requirements for those with special needs because of physical or other disabilities are described by experts as are the accessibility features of resources they may use. The latter are usually the result of working with software that in most cases now produces a

single report in Evaluation and Report Language (EARL). It would be possible to make many statements from this single report but there does not seem to be any value in doing that. As the statement itself is encoded, it seems much cleaner and neater to have a single value for the element, a URI, and have all the information together. In many cases, the value that would need to be put into the existing DC elements and refinements would not be human-readable.

4.3.1.4. Needs

4.3.1.4.1. Is there a clear requirement in existing implementations for the term in support of resource discovery?

There is now a legal requirement in many countries that content should be accessible. This is the first step towards making it possible for users because it identifies problems and provides for systems that can detect metadata warning of mismatches and look for content that is accessible to replace or supplement the original inaccessible content.

Systems that are used for publishing resources in situations where accessibility is required (esp educational settings) are often not able to satisfy regulations with a single resource and so need a way to discover and combine resource compnents, to augment and swap them from distributed sources or cumulatively author them, and for all such implementers, this adaptability information is proving to be what they want. System developers in Europe and north America are already using the user needs and preferences information to match resources to users and the Assistive Technology Indistry Association has supported it and the USB standards maintainers are also supporting it.

4.3.1.4.2. Is there a demonstrated need for the proposed element or element refinement?

Yes. Currently it is not possible to determine if content will be accessible in advance of it being delivered. This can cause problems for a person who cannot access the content, but it also is not the goal-state that is of interest to those trying to ensure accessibility for their users. This element does not involve claims of accessibility or otherwise which might better be described as conformance metadata. It is very important to distinguish this element from such an element because of the legal implications of inaccessibility and therefore, expected resistance to the use of such an element.

Microsoft identified in research in 2004 that >60% of working adults in the US have problems with accessing digital resources even when they have suitable equipment and telecommunications systems and software. Many of these problems can be solved by resource adaptation but without the proposed descriptions of the resource, even a fully adaptable resource will not be discoverable or adaptable.

4.3.1.4.3. Are there existing implementations or encoding schemes, etc., which use the term?

Yes. The lead has been taken by the Assistive Technology Resource Centre's work at the University of Toronto. This work has been undertaken with the Canadian Government. It is of interest to the International Standards organization and is currently being considered for endorsement by them. The exemplary implementation of The Inclusive Learning Exchange (TILE) which can be tried at http://inclusivelearning.ca/). The IMS Global Project have adopted the term.

4.3.1.5. Fits with other DCMI-maintained terms

4.3.1.5.1. Follows existing principles of refinement

4.3.1.5.2. Is well-formed

Yes

4.3.1.5.3. Does not conflict with or create ambiguity with regard to existing DCMI-maintained terms

4.3.1.5.4. Does not create problems for existing legacy implementations if those implementations have followed recommended practice.

Decision Tree Table

Condition 1:

Can the need be solved with a vocabulary encoding scheme for an existing DCMI Element or Element Refinement?

Because a number of things need to be recorded and they would fit into a number of different elements, and because the values are not really useful unless they are in machine-actionable form and so new vocabularies might be needed, it would not really be practical or sensible to squeeze adaptability into the existing structure.

If so, do that; else …

Condition 2:

Can the need be solved through an application profile that references an element or element refinement from an existing and recognized non-DCMI namespace?

Again, this would be possible. DCMI could have a simple application profile that does no more than reference the IMS elements etc. This would be difficult but it is not what a number of people have expressed they want and it would not contribute to the work of promoting accessibility. It seems that a single element, easily included in a DC set, is the best way to encourage the huge DC-using community to embrace adaptability.

The critical thing is that the DC-Accessibility Working Group participated in the development of the element that it is now asking to have endorsed by the Usage Board. Although there is IMS ‘ownership' and maintenance of the early documents, in fact they are as much DC documents as they are IMS documents. It is likely that the new DC term will be used extensively by communities concerned with adaptability including education. mobile, device independent and other communities.

If so, do that; else …

Condition 3:

Can the need be solved with a new refinement for an existing DCMI element?

The answer to this is the same as for the element, above.

If so, do that; else …

Condition 4:

Create a new DCMI Element (and, if necessary, Element and Vocabulary Encoding Scheme) to meet the need.

Overview

The DC Accessibility Working Group has been investigating issues related to accessibility and metadata for four years. At the beginning, it seemed that it would be easy to find some values for existing DCMES terms and perhaps a profile should be specified to make this process easy. After some analysis, however, it became clear that this approach would not be simple and in fact would require the introduction of a number of new refinements to some of the elements. Meanwhile, work in the accessibility field in other contexts showed that there was resistence to metadata that would incriminate content authors and publishers and so the idea of using the approach of metadata to represent conformance to standards or accessibility specifications was recognised as likely to be unpopular.

More recently, major research has shown that comformance to accessibility specifications does not guarantee accessibility for individual users.

In 2003, work on a new approach was adopted. It focused on the matching of resources to users and avoided issues of conformance. The test of accessibility is now whether a resource, as presented is accessible to a user. It serves all users who at various times may find themselves, for one reason or another, unable to use resources as they are. These users need to adapt the resource to have sound redered as text, for example, because they cannot hear sound at the time of resource delivery. This may mean that a user gets access to a resource only after it has been adapted to suit their sensory modality or other needs.

The work, while initiated in a forum hosted by the IMS Global Learning Consortium, was specifically undertaken with the intention of being collaborative work with some other groups but primarily the Dublin Core Accessibility Working Group.

Dublin Core Accessibility Working Group members were invited to participate, as some did, and all the work was specially documented according to Dublin Core practices so that DC Accessibility WG members could participate. This involved a special effort on the part of the IMS Consortium. There was also a special effort to ensure that the work was internationally acceptable and so several fora beyond the open IMS one were supported so that discussion could be open. The fluidity of association of the working participants was perhaps unusual but within the accessibility world, this appears to be the best that can be hoped for at present.

The following communities were engaged in discussions about the new element:

- DC Accessibility WG
- IMS Global Learning Consortium Accessibility Special Interest Group
- W3C Web Accessibility Initiative Interest Group
- EuroAccessibility
- SIDAR Foundation
- WGBH-NCAM
- INCITS V2
- CEN-ISSS APLR
- CEN MMI-DC
- ISO JTC1 SC36

The approach adopted was based on the information about users and resources that is necessary to match resources to an individual user's needs and preferences. It involved developing a formal approach to issues of accessibility that should be embedded in metadata. It is not relevant why a user has the needs: it may be because of the context in which they are using the resource, it may because of the device they are using or it may be because they have a disability such as aural impairment.

A profile of categories of causes of mismatches between users and resources was developed and formally defined in a specification that became an IMS specification. This profile involves three categories: control, display (or presentation) and content. The development of this profile was undertaken with full awareness that it would not just be useful in the educational community where it was initiated but that it would be good for all situations. It was documented in a way that would make it suitable for use within Dublin Core style systems if user profiles were being used. The user needs and preferences description was called the AccLIP Profile and published at http://www.imsproject.org/accessibility.

There have been several efforts to broaden the DCMI approach to user descriptions, suggesting that DCMI has concentrated on resource description but that user descriptions may also be well-described using DC architecture. A short paper on this was presented at the DC 2004 Conference in Shanghai. Another suggesting that such a profile need not be seen as a description of a person but simply as a set of requirements has been submitted for the DC 2005 conference.

The second part of the work involved developing a mirror specification that could be used to describe resources to enable the matching process. This work was undertaken more formally in collaboration with the DC Accessibility Working Group. The specification was developed with emphasis on the need for it to be immediately useful for the Dublin Core community and applications. This work has now been completed and is documented as the AccMD Profile and published with an Overview, Best Practice Guide, Information Model, XML Schema and Binding at http://www.imsproject.org/accessibility.

Meanwhile, there have been many suggestions and contributions to the process. One significant one has been to show the relationship between the work as originally undertaken. Then, it was to support people with special needs. Now, it supports everyone when they have special needs by describing adaptations needed by users and available for resources. As it is proposed that the term should be used to describe characteristics of a resource that are necessary if the resource is to be presented on a mobile phone, transformed from text to speech, or delivered without images or other visual content, the best term name is probably 'adaptability'. The characteristics are about the adaptability of a resource. Those with what are known as 'accessibility' needs are not distinguished but their needs are included. This is an inclusive approach to delivery of digital resources. What is to be described using this term is a quality of the resource that is not described in any other DC term, its adaptability.

The element being proposed and its use within the parallel IMS Gloobal Learning community, are described in the published IMS Overview and Best Practices Guide. These documents include explanations of how the element can be used and a set of FAQs designed to give a sense of its context and value. This information is not repeated here (please see http://www.imsproject.org/accessibility).

The work was implemented by the Canadian participants in the work and can be tried on the demonstration site http://inclusivelearning.ca/. The implementation is in a system known as TILE, The Inclusive Learning Exchange. It uses both the user requirements profile and the resource metadata profile.

The best value for the proposed element will probably be a URI that points to a machine-readable statement. A typical example will be inclusion of the new W3C 'MobileOK' mark (an image with a pointer to a URI.

EARL, the Evaluation and Report Language developed by W3C, is an XML-based constrained language that extends the RDF (Resource Description Framework) family. An EARL statement not only contains a simple statement of a resource and associated property, in a machine –readable format that includes a way of resolving the property type, but it mandates that there will be at least three such properties, including one that identifies the creator and another that identifies the date of the statement. EARL statements are recommended for descriptions of digital learning resources by IMS.

The use of an EARL statement by the accessibility community might seem like a heavy requirement for the ultimate value of an element. This is not considered to be the case because within the accessibility world, EARL statements are treated as typical good practice and many of the tools that are used to test accessibility produce them. Sometimes the EARL statements are produced in order to record the adaptability of a resource in terms of conformance to some standard or specification, but in this case the required information will be available for the element as well, at least after a simple transformation.

The use of an EARL statement is equivalent to the use of ‘controlled' vocabularies. It is to be recommended but not essential. There is already an IMS XML schema that defines the values to be used and the use of XML conforming to this schema will be sufficient. The XML schema is available at http://www.imsproject.org/accessibility.

DC Accessibility Working Group
10 May 2005

DC-Accessibility Element Proposal: Accessibility

Title:	Working Group Term Proposal: Adaptability
Creator:	Dublin Core Accessibility Working Group
Date Issued:	2004/8/28
Identifier:	http://www.ozewai.org/prop-reqs-table2.html
Replaces:	None
Is Replaced By:	None
Latest Version:	http://www.ozewai.org/prop-reqs-table2.html
Status of Document:	Proposal to DCMI Usage Board
Description of Document:	This document presents a proposal from the Dublin Core Accessibility Working Group for a new element named "Adaptability"

Proposal

Name:	http://purl.org/dc/terms/adaptability
Label:	Adaptability
Definition:	A statement describing characteristics of the resource that affect how it can be adapted so it can be perceived, understood or interacted with by users
Comment:	An Adaptability description might be used to match a (digital or physical) resource to a description of user or user agent needs and preferences.
Examples:	The following example is extracted from the IMS documentation: **Scenario:** An HTML file contains text and an embedded Flash animation (visual only, no sound). There is also alternative textual content to the animation defined by accessibility meta-data as an equivalentResource containing alternativesToVisual properties. A user profile has a content element with the alternativesToVisual preference set and wishes to interact with the aggregate file. The system applies the matching test on the aggregate HTML resource and sees it has a hasVisual property with a value of true. Subsequently it sees the animation has an equivalentResource with an alternativesToVisual which matches the user's content preferences. At this point the system replaces the animation with the text alternative. The system modifies the aggregate resource by changing its reference to the animation to a reference to text, i.e., the embedded Flash animation's <object> tag is replaced with a <p> tag containing the alternative textual content. **Other examples**
Type of term:	Element

Term qualified:	None
Why needed:	The element will be significant in the case of a user with special needs such as an inability to use particular sensory modalities at the time of delivery, or who uses a particular devie that has limitations such as a mobile device with a very small screen. Some users have detailed requirements such as that they cannot use resources that use redand green to convey informative content. Such users and their agents will find the descriptions in the term useful either to allow them to confirm or reject presentation of the resource or to discover substitute content to replace or augment content that is a problem. Note that sometimes what is inaccessible is only a small part of what might be thought of as a composite object, such as an image in a web page. This object in its original form is known as a primary object and may within itself contain other forms of the same object, known as alternative objects. Where there is an alternative object that is not part of the primary object, it is associated with the primary object as an equivalent object but has separate metadata as it is a separate object. There can be little doubt that metadata that allows a user to find a resource that is accessible to them is always a high priority. The values of the proposed element can be critical to a user's access to a resource as it wil enable the adaptability of the resource to their needs or the needs of their access devices.
Working Group/ community support:	The proposed element has been developed over several years. The range of problems that a resource may have for users was first analysed and described. This process was widely advertised and subjected to public debate and comment by the collaborating group, the IMS Global Learning Consortium. Dublin Core Accessibility Working Group members were kept informed of this process and invited to comment on the developments and drafts, and they did. Discussion took place on a number of discussion lists, primarily the IMS coments list. All comments sent to IMS were formally processed. The proposed element is now undergoing more scrutiny in the ISO JTC1 process where it has reached the fisrst draft stage. To review the discussion archives of the Accessibility Working Group, see http://www.jiscmail.ac.uk/cgi-bin/wa.exe?A0=dc-accessibility. The proposed element is a reflection of the user profile noted above. It also has been developed by an international collaborative team containing many DC Working Group members and in full knowledge of its development being posted regularly to the DC Accessibility Working Group. IMS again managed the process of advertising the profile, soliciting and managing comments and issues that were raised.
Proposed status:	Recommended
Related DCMI terms:	The possibility of capturing and recording necessary metadata relevant to adaptability to support accessibility has been considered over several years. Although the DCMES has a number of elements and element refinements that individually might be suitable for inclusion of the information being proposed, this approach was considered by the Working Group and relevant communities and rejected.

Related non-DCMI terms:	IMS AccLIP and AccMD (AccessibilityForAll User Profile and Accessibility Metadata) These two specifications are those worked on by DC Accessibility WG members as well as IMS participants. They provide the information model and recommendations for encoding of the information but they are to be used within IMS Content packaging. This was always understood as different from their being DC used within a DC term and they were specifically developed with their DC use in mind.
Impact on applications:	Minimal. Since current DC-based applications provide no conflicting means of unambiguously referencing accessibility profiles, impact on those applications would be minimal.
About the proposers:	Scope of DC Accessibility Working Group: The DC Accessibility Working Group has been engaged with the issue of developing metadata for accessibility since its inception. Members of the Working Group are involved in a range of accessibility activities in a range of countries and have endeavoured to work collaboratively across all communities. Aims: To maximise opportunities for accessibility for all users where accessibility is defined as the matching of resources in terms of control, display and content to user needs and preferences. Specifically, this will cater for the needs and preferences of those with disabilities. Brief History: Current status: A collaborative exercise was initiated in which the DC Accessibility Working Group worked with others to discover the best way to specify accessibility metadata. The IMS Accessibility Working Group, as part of the IMS Global Learning Consortium, hosted the work and will provide the ongoing support and publication of the relevant specifications on behalf of the collaborators who specifically include the DCMI. The IMS Accessibility Working Group has proposed the element to the IMS Technical Board and it has been accepted as an IMS recommendation. Pointer to IMS Reports, Documents and Discussion Archives: http://www.imsproject.org/accessibility.

Examples of How to Use the DC:adaptability Term

In HTML

If the resource is composed of multiple mixed adaptability features then multiple or repeated Adaptability elements should be used to describe the main components.

Using free text:

```
<head>
<meta name="DC.description" content="HTML file with embedded video" />
<meta name="DC.adaptability" content="this resource has visual content" />
</head>
```

Using controlled vocabulary:

```
<head>
<meta name="DC.description" content="HTML file with video" />
<meta name="DC.adaptability" scheme="IMS-AccessForAll" content="has modality; type=visual" />
</head>
```

Using a pointer to an EARL statement:

```
<head>
<link rel="DC.adaptability controlFlexibility" href="http://www.example.org/flex.rdf" />
</head>
```

Other values taken from the controlled vocabulary that might be used:

- `Adaptability="has modality; type=audio"`
- `Adaptability ="has modality; type=visual"`
- `Adaptability ="sign language; lang=French-LSF"`
- `Adaptability ="caption track; lang=en; type=verbatim; rate=180 WPM"`
- `Adaptability ="audio description; type=standard"`
- `Adaptability ="alternative text; lang=de"`
- `Adaptability ="graphic alternative; lang=bliss"`
- `Adaptability ="color avoidance; colors=red,green"`

Note: The simplest example recommended is a statement of the type `“has modality; type=visual”`.

This description will alert any application looking for standard format information to the need to investigate further the ways in which this resource can be presented to a user who requests it but at the time cannot use visual sensory perception.

Another free text example is `“This resource may not be accessible if a user does not have auditory capabilities at the time but the publishers will supply captions and a transcript upon request to help@publisher.com.”`

Such an example contains some information that can be useful but, in general, it will not be useful for triggering machine actions such as the seeking of other equivalent resources. Controlled vocabularies are recommended to facilitate interoperability and machine-readability.

Primary and Equivalent Relationships

In the AccessForAll information model, there is the concept of 'primary' and 'equivalent' resources. The primary resource is the original resource. It can contain a set of components from which a composition of the resource to be delivered is assembled. These components provide alternative access modalities and have their own adaptability properties. An equivalent resource is another resource or resource component that is used to provide a user with a version of the original resource they can use either by being substituted for the original resource or a component of it (alternative) or by augmenting the original resource or a component of it (supplementary).

Continuing with the DCSV format.

Examples:

In the description of the primary resource where there is a known equivalent available:

```
Adaptability ="primary; equivalentResource=http://www.example.org/equiv.txt"
```

In the description of the equivalent resource:

```
Adaptability ="equivalent; type=supplementary; primaryResource=http://www.example.org/image.
foo"
```

Using DC expressed in XML:

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

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

<dc:title>dc:adaptability Example</dc:title>

<dc:description>Equivalent movie with captions </dc:description>

<dc:adaptability xsi:type="dcterms:IMS-AccessForAll">caption track; lang=en; type=verbatim;
rate=180 WPM</dc:adaptability>

<dc:adaptability xsi:type="dcterms:IMS-AccessForAll">equivalent; primaryResource=http://www.
example.org/prim.mpg</dc:adaptability >

</metadata>
```

XML with a pointer to an EARL statement

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

<metadata xmlns="http://example.org/myapp/" xmlns:dc="http://purl.org/dc/elements/1.1/">

<dc:adaptability>displayTransformability; scheme=IMS-AccessForAll; location=http://example.
org/something.rdf
</dc.adaptability>

</metadata>
```

Specification conformance

DC Adaptability is not about conformance to an 'accessibility standard' (that would be a DC relation element value) but there are specifications that relate to aspects of what is relevant to DC Adaptability. In the case where there is a document that provides machine-readable information about conformance of a resource to such a specification, some or all of that machine-readable information may be of interest, as will be the case where the specifications are the W3C WCAG specifications.

DC metadata in RDF

Some IMS data included directly

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"

xmlns:dc="http://purl.org/dc/elements/1.1/" xmlns:dcterms="http://purl.org/dc/terms/" xmlns:
I="http://purl.org/dcterms/IMSAccessTerms/">

<rdf:Description>

<dc:adaptability>

<rdf:Description rdf:about="#desc1">

<rdf:type rdf:resource="http://purl.org/dcterms/IMSAccessTerms/CaptionTrack" />

<dc:language>en</dc:language> <I:captionType rdf:resource="http://purl.org/dcterms/
IMSAccessTerms/Verbatim" />

<I:speechrate rdf:datatype="http://purl.org/dcterms/IMSAccessTerms/WPM">180</I:speechrate>

</rdf:Description>

</dc:adaptability>

</rdf:Description>

</rdf:RDF>
```

Note that this uses serious RDF, although it could be further simplified if there are formal Class/subClass definitions. (For example anything called type is likely to be a candidate for simplification using RDF Vocabulary Language.

RDF linking to an external resource

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:dc="http://purl.org/dc/
elements/1.1/">

<rdf:Description>

<dc:adaptability rdf:resrouce="http://example.org/resultstore/someResult" />

</rdf:Description>

</rdf:RDF>
```

Text transformation

In the case where there is text, the ‘transformability’ of the text may be an issue. Determining this depends upon answering a series of questions about the text. There are already tools that automate the process. They are often used to determine conformance to accessibility specifications. These tools can produce EARL statements that contain information needed by machines to determine if they can do text transformation and how to do it. A pointer to such a statement would be used as the value for textTransformability of a resource or component.

Proposal for DCMI Extension Namespaces

Creator: Thomas Baker
Contributor: Makx Dekkers
Date: 2005-05-04

About this document

The DCMI Board of Trustees, at its meeting in Seeheim, Germany, in April 2005, revisited DCMI's mission statement, extending the mission explicitly to encompass resource description. In light of this revised mission, the Trustees also discussed the need of DCMI Working Groups to find a home for descriptive and other, application- or domain-specific terms that they define in the course of developing their Application Profiles.

The absence of such a home is perceived as a crucial bottleneck delaying the forward progress of Working Groups and the practical adoption of Application Profiles. At the same time, the DCMI Board of Trustees felt there was an opportunity for DCMI to increase its value to the community by offering a solution to this need, also acknowledging that care needs to be taken to do this in a sensible and sustainable way.

The Board of Trustees tasked the DCMI Directorate to elaborate a proposal to address these issues and identify the actions that need to be taken to establish a home for such terms as a service to the community.

This proposal is work in progress, to be revised after discussion within the DCMI Directorate, Board of Trustees, Usage Board, and Advisory Board.

Caveat: This document refers to the notion of "namespace" in line with the existing DCMI Namespace Policy [5]. As discussed in the Usage Board and Architecture Working Group, however, this use of "namespace" has been seen as needing clarification. Prior to implementation, the name of the proposed service should perhaps be reconsidered in light of such a clarification.

Background: the current situation

The mission of DCMI has long emphasized resource discovery across domains. Accordingly, the DCMI Usage Board has used "cross-domain validity" and "usefulness for resource discovery" as twin criteria for evaluating terms proposed for inclusion in the set of DCMI Terms [1,12,13]. Terms approved for inclusion are assigned URIs formed on the basis of three base URIs [2,3,4] and maintained in accordance with the DCMI Namespace Policy [5].

DCMI Working Groups, on the other hand, typically define their task as one of developing an "Application Profile" for use in a particular domain or application, e.g., for describing government information, learning materials, library resources, or resource collections. In order to do this, Working Groups often need to coin new terms for descriptive needs unmet by existing vocabularies.

To the extent that such additional terms are not demonstrably useful both for "resource discovery" and "across domains", they have hitherto not been considered good candidates for inclusion in the central set of DCMI Terms. Some such terms have been rejected by the Usage Board for not meeting these criteria, while others have been approved on the basis of a relaxed interpretation of the criteria. Some terms of a

clearly domain-specific or purely descriptive nature were simply never proposed.

Until now, the option available to Working Groups developing such terms has been to declare them in non-DCMI namespaces. Guidelines laying out options for doing so have been drafted [6]. However, finding an appropriate home for new terms -- e.g. an institution which can at a minimum guarantee that URIs will not be reassigned and will remain resolvable to documentation on the Web for the foreseeable future -- has proven to be a significant hurdle.

At the same time, the Usage Board has developed and begun to test processes for reviewing Application Profiles [7,8,9]. The DCMI Abstract Model [10], which describes the nature and composition of "DCMI metadata descriptions", was approved as a DCMI Recommendation in March 2005 and provides a solid and agreed basis for such reviews. In principle, Usage Board review is aimed at establishing whether a given application profile for such descriptions is in conformance with the Abstract Model.

Proposed approach

In order to be able to provide a home for terms outside of the criteria governing the inclusion in the core set, DCMI needs to take a number of actions and formulate a number of policies. The Directorate proposes the following approach:

1. To host namespaces for terms outside of the core set under the name "DCMI Extension Namespaces".

2. To define and describe DCMI Extension Namespaces in a revised and extended DCMI Namespace Policy [5].

3. To assign a base URI for DCMI Extension Namespaces and formulate a procedure by which new terms will be assigned URIs -- also as an extension to the DCMI Namespace Policy [5]. The proposed base URI for DCMI Extension Namespaces is http://dublincore.org/extension/.

4. To formulate guidance on the naming and identification of terms in DCMI Extension Namespaces to the extent that these topics are not already covered by the DCMI Namespace Policy [5] and the DCMI Policy on Naming Terms [11], and to formulate and document a model for versioning historical descriptions of DCMI Extension Namespaces.

5. To develop and clarify the criteria, formal processes and workflow by which a DCMI Extension Namespace is created, terms are reviewed and approved, approved terms are added to that namespace, and a DCMI Extension Namespace, once created, can be edited or changed over time. This involves describing the roles of a DCMI Strategic Activity (i.e. a DCMI Working Group) and the Usage Board in the creation and review of terms for a DCMI Extension Namespace.

6. To formulate a policy to define and clarify the persistence of DCMI Extension Namespaces and the maintenance responsibility that DCMI takes towards these namespaces, in particular clarifying the respective roles and responsibilities of the DCMI Usage Board in the medium- and longer-term maintenance of DCMI Extension Namespaces given that DCMI Strategic Activities will usually have a limited lifespan.

7. To define the notion of a DCMI Strategic Activity, involving the description of any duties or obligations between DCMI as an organization and the leaders or members

of a DCMI Strategic Activity, along with any clarifications regarding membership and operational procedures of Strategic Activities. This may involve obtaining commitment for the management of the Strategic Activity from a institutional stakeholder, possibly a DCMI Affiliate.

Discussion

In recognition of the long-term maintenance and workload issues raised by the prospect of additional DCMI-owned namespaces, it is suggested that access to such namespaces be limited to activities (i.e. Working Groups) designated by DCMI as "strategic activities". In this approach, a DCMI Strategic Activity can present an Application Profile to the Usage Board for review for conformance to the Abstract Model, and any new terms in a profile judged to be "conforming" would be eligible for declaration in a DCMI Extension Namespace. The Collection Description Working Group has been suggested as a test case for the status of DCMI Strategic Activity.

It seems reasonable to build in safeguards to avoid that DCMI be confronted with the obligation to maintain a rapidly growing number of additional terms over time. Firstly, the Usage Board, currently chartered as a central, generically-scoped committee, cannot be presumed to have all the necessary expertise to maintain terms that may be application- or domain-specific. Secondly, care should be taken not to create an overwhelming workload for the Usage Board in maintaining potentially rapidly growing set of terms.

It can be expected that terms proposed for DCMI Extension Namespaces will typically be terms "around the edges" of Dublin Core -- the "missing pieces" of the particular Application Profile under development. A DCMI Extension Namespace might typically have between one and a dozen or so properties. In addition, a DCMI Extension Namespace may hold terms for small controlled vocabularies of values. These terms most likely will only make sense in the context of the Application Profile for which they were created. Therefore, DCMI should put more emphasis on Application Profiles -- by assigning them status, describing them on Web pages, and featuring them as examples of good practice -- than on DCMI Extension Namespaces per se.

References

[1] http://dublincore.org/documents/terms/
[2] http://purl.org/dc/terms/
[3] http://purl.org/dc/dcmitype/
[4] http://purl.org/dc/elements/1.1/
[5] http://dublincore.org/documents/dcmi-namespace/
[6] http://www.ukoln.ac.uk/metadata/dcmi/term-identifier-guidelines/
[7] http://dublincore.org/usage/documents/profiles/
[8] http://www.dublincore.org/usage/documents/process/#six
[9] http://www.dublincore.org/usage/documents/process/#conforming
[10] http://dublincore.org/documents/abstract-model/
[11] http://dublincore.org/documents/naming-policy/
[12] http://dublincore.org/usage/documents/process/#recommended
[13] http://dublincore.org/usage/documents/process/#conforming
[14] http://www.w3.org/2004/02/skos/core/spec/
[15] http://www.jiscmail.ac.uk/cgi-bin/wa.exe?A2=ind0202&L=dc-usage&F=&S=&P=3628
[16] http://dublincore.org/usage/meetings/2004/10/ISSUES/dcx/

Appendix: Usage Board discussions about "namespace hosting"

The problem of declaring and maintaining terms outside of the central vocabularies has been discussed at various times by the Usage Board. In January 2002, the position of the Usage Board was as follows [15]:

> When the Usage Board approves a term, that term will go into the dcterms namespace (or perhaps into a new namespace in some exceptional circumstances). If the UB does not approve a term, then projects will have to create 'a new (non-DCMI) namespace' -- which means a namespace with a non-DCMI namespace URI, which means that it isn't hosted on the DCMI site.
>
> In order to avoid confusion, such non-DCMI namespaces should not be allowed to carry 'DC' in their name.
>
> Views of non-DCMI namespaces may well be available thru the DCMI Registry, but the implications for review process and registry interface have yet to be worked out.

In subsequent discussion between the Usage Board and the Directorate, it was agreed that DCMI as an organization (and not the Usage Board in particular) might make a strategic decision to offer Namespace Hosting as a service to working groups as long as care were taken to ensure that the distinction between DCMI Term namespaces and a hosted namespace were never ambiguous (for example, by avoiding the use of "DC-" in the title of a namespace). It was agreed that hosted namespaces would be outside the jurisdiction of the Usage Board inasmuch the Usage Board would not be responsible for responding to requests for editorial changes to hosted terms.

At its meeting of May 2002, the Usage Board pictured this as a model consisting of a core (the Dublin Core), a semi-periphery of additional DCMI terms, and a periphery of non-DCMI terms identified under the domain of a yet-to-be-created, DCMI-owned but non-DCMI-branded namespace host.

For several reasons, DCMI did not implement this idea at the time. It was not clear by whose authority, if not the Usage Board's, terms were to be declared in a hosted namespace. Working groups are by definition temporary, so maintenance responsibility, to the extent it were implied, would by default revert to DCMI as a whole, and therefore to the Usage Board. It was also recognized that a DCMI-hosted namespace would offer a solution attractive for many working groups, generating a potentially unmanageable demand for access to the namespace host.

Furthermore, it was not clear by what criteria and processes DCMI could -- if not via the Usage Board -- regulate such access and ensure at least a minimal quality of terms declared in a hosted namespace. Nor was it clear by what processes or criteria DCMI might assert an implied right to deprecate or even remove terms proven by implementation to be problematic.

In a nutshell, the prospect of creating a DCMI-owned namespace host in the absence of a clear process model seemed to risk associating DCMI with a potentially large number of unmaintained (and unmaintainable) namespaces. Taken individually, these namespaces would look fragmentary -- the "missing pieces" of this or that application profile. Collectively, they would present an incoherent whole.

(One recurring variant of the Namespace Hosting idea has been for DCMI to create a namespace where working groups can "park" new or experimental terms without going through a formal approval process in order to make new or experimental terms citable by URI, thus supporting a less "bureaucratic", more bottom-up, market-driven approach to growing vocabularies. However this scenario, according to which working groups would get direct access to a namespace

host, seems even more problematic than others. This variant is discussed in [16].)

After periodic consideration of alternatives, the Usage Board has always circled back to the notion that in the end, terms must be maintained, and it should always be clear who is maintaining the terms -- at any rate for all terms which have or could be perceived to have branding by DCMI. In the absence of alternative solutions within DCMI, the Usage Board has followed the conservative course of limiting its vocabulary management activities to a small and slowly growing vocabulary of terms in accordance with reasonably strict policies, processes, and principles.

Some Issues Related to DCMI Extension Namespaces

2005-05-13

-- "Documentation clutter": If terms are going to "expire" (i.e. cease to be actively maintained) at some point, we would want to have some way to present a view of the "fresh" terms.

-- Maintenance: The SKOS vocabulary is maintained directly in RDF, which is used to generate Web pages. We should perhaps look into this as a method for maintaining EXT-NS terms, or perhaps even share tools with the SKOS community.

-- Scope Statement: There is an old action on Andrew to draft a Scope Statement: what are the boundaries and criteria for inclusion of terms in DCMI (i.e. what 'resource discovery' and 'cross-domain' means in practice), to be included in UB Mission and Principles, UB Process, or elsewhere. Is this still relevant?

-- Status of "Conforming": In March 2004 in Bath, we reaffirmed that the Usage Board can assign the the status of "conforming" to an Application Profile based on a significantly more thorough review focused on elements and element refinements at the point of review. The AP designated as "conforming" (i.e., a snapshot of the AP document at the time reviewed) would be archived on the DCMI Website. Changes to the AP should result in a new AP and resubmission to the UB (i.e., for new "time stamp").

-- To batch or not to bach. The DCMI Extension Namespaces proposal is deliberately in the plural - "namespaces". The idea is that batches of terms will be subject to maintenance by different communities or organizations. Putting all terms into one big would make it less convenient to divide up maintenance responsibility.

-- Naming (URIs) - base URI. The DCMI Extension Namespaces need a base URI; the current preference is to use something based on http: //dublincore.org/, such as http://dublincore.org/extension/ or http: //dublincore.org/ext/. However, the argument has been made that, for consistency, http: //purl.org/ should be used.

-- Naming (URIs) - sub-namespace URI. Given that the DCMI Extension Namespaces will be subdivided into "sub-namespaces" with separately maintained batches of terms, then the question is how to name those batches. In theory, URIs are opaque. In practice, however, people do try to interpret the strings used in a URI. We should therefore be careful to choose names that do not convey misleading notions. Some possibilities:

   a) To use strings such as "dccdwg", as in: http://dublincore.org/ext/dccdwg/.

   b) To assign a "time-stamped" base URI to a working group, e.g., http://dublincore.org/extra/2005/09/ for Collection Description WG. Advantage: avois strings which reflect the origins of a batch of terms in a particular working group or activity.

   c) To number the extension namespaces sequentially:
      http://dublincore.org/ext/01/ - for collection description
      http://dublincore.org/ext/02/ - for libraries

-- Use of the term "Namespace". DCMI needs to be clear as to whether a DCMI Namespace is "a collection of URIs" (i.e., the generally accepted notion that a "anemspace" is a "set of names") or "a collection of terms identified by URIs" -- and at any rate not an XML namespace. DCMI -- but not necessarily the Usage Board -- would need to use the term "namespace" consistently in the Abstract Model and the Namespace Policy. For example:

> DCMI namespace
>
> A collection of 'term names' (i.e. 'term URIs') where each term is assigned a URI that starts with the same 'base URI'. The 'base URI' is known as the 'DCMI namespace URI'. (Note that a 'DCMI namespace' is not the same as an 'XML namespace').
>
> Note that the grouping of 'term names' into a 'DCMI namespace' is orthogonal to the grouping of 'terms' into a 'DCMI vocabulary'. 'Term names' are grouped into 'DCMI namespaces' in order to ease the assignment of URIs to 'terms' and to streamline their use in particular encoding syntaxes. 'Terms' are grouped into 'DCMI vocabularies' in order to meet a functional need.

Title: DCMI Usage Board Review of Application Profiles
Creator: Thomas Baker
Identifier: http://dublincore.org/usage/documents/2003/02/11/profiles/
Latest version: http://dublincore.org/usage/documents/profiles/
Date modified: 2003-02-11
Description: This document defines the term "Application Profile" in the context of the Dublin Core Metadata Initiative. Criteria for Usage Board review of Application Profiles and guidelines for submission are outlined in the DCMI Usage Board Administrative Processes document [PROCESS].

"Application Profile" defined

For the purposes of DCMI Usage Board review, an Application Profile (AP) is a declaration of which metadata terms an organization, information resource, application, or user community uses in its metadata. Moreover:

-- By definition, an AP cannot "declare" new metadata terms and definitions; it only "reuses" terms from existing element sets [HEERY].

-- The ideal element set will use URIs to uniquely identify its terms within XML namespaces [DCMI-NAMESPACE]. As of 2002, however, this cannot be required.

-- By definition, any new term coined for use in an AP must first be declared in a form citable in the AP.

-- An AP may also provide additional documentation on how the terms used are constrained, encoded, or interpreted for particular purposes.

As of 2002, APs are seen primarily as a form of documentation, the purpose of which is to help implementor communities harmonize their metadata practice. It is hoped that in the longer term, machine-processable versions of such APs based on data models such as RDF will provide a basis for automating metadata interoperability functions such as semantic crosswalks and format conversions.

References

[DCMI-NAMESPACE] Andy Powell, Harry Wagner, Stuart Weibel, Tom Baker, Tod Matola, Eric Miller, Namespace policy for the Dublin Core Metadata Initiative, http://dublincore.org/documents/dcmi-namespace/.

[HEERY] Rachel Heery and Manjula Patel, Application profiles: mixing and matching metadata schemas, Ariadne 25, September 2000, http://www.ariadne.ac.uk/issue25/app-profiles/intro.html.

[PROCESS] http://dublincore.org/usage/documents/process/.

Excerpt from DCMI Usage Board Process (full text of which
is elsewhere in this packet)

6. Proposals for Registration of Application Profiles

6.1. Sources of proposals

6.1.1. DCMI working groups

6.1.1.1. Existing working groups or working groups established for the purpose of developing proposals

6.1.2. Metadata implementers

6.1.2.1. Established projects or research groups

6.1.3. UB itself

6.2. For the purposes of review by the Usage Board:

6.2.1. The Usage Board is interested in reviewing application profiles that make substantial use of Dublin Core elements. The review of application profiles by the Usage Board serves to:

6.2.1.1. analyze the usage of Dublin Core within significant implementations;

6.2.1.2. assign a DCMI stamp of approval;

6.2.1.3. promote the sharing of application profiles between communities; and

6.2.1.4. identify new terms as candidates for inclusion in DCMI namespaces.

6.2.2. Application profiles must provide, for each term, an identifier of the element set where it is defined, ideally in the form of URIs for individual terms.

6.2.3. If the terms in an application profile describe anything other than generic "resources" (the typical domain of Dublin Core), the application profile must make this clear. This is particularly important if an application profile is based on a data model that describes multiple classes of resources, such as agents or collections.

6.2.4. It is recommended that application profiles be prepared using the Dublin Core Application Profile guidelines published by CEN [http://www.cenorm.be/cenorm/businessdomains/businessdomains/isss/cwa/cwa14855.asp].

6.2.5. Each application profile must provide, or point to, a short text that describes:

6.2.5.1. The context and purposes in which the application profile is used or is likely to be used.

6.2.5.2. The organizations or individuals involved in its development and a capsule history thereof.

6.2.5.3. Any arrangements, policies, or intentions regarding the future development and maintenance of the application profile.

6.3. Review of Application Profiles by the Usage Board

6.3.1. An application profile is "well-formed" if it is

presented in accordance with the broad and flexible requirements outlined above. These presentation requirements may become more specific as "good practice" emerges over time.

6.3.2. Usage Board review focuses on the use of terms related to Dublin Core terms and on any data models that provide a context for those terms. The Usage Board is agnostic about the use of terms not directly related to Dublin Core; strictly speaking such terms are outside the scope of Usage Board review.

6.3.3. The use of terms related to Dublin Core (such as refinements of Dublin Core elements, or Dublin Core elements that have been constrained for particular contexts) will be evaluated from the standpoint of semantic conformance, grammatical principle (eg, "dumb-down"), clarity, and good practice.

6.4. Publication and use of Usage Board Reviews

6.4.1. An application profiles that "pass" review will be assigned the status of 'conforming'.

6.4.1.1. The status of 'conforming' indicates a Usage Board assessment of the application profile as of the date of its submission for review.

6.4.1.2. Changes to already 'conforming' application profiles require further Usage Board review of the application profile in whole or in part according to the processes and criteria outlined in sections 6.1 through 6.3.

6.4.2. For application profiles that "pass" review, the Usage Board will publish a Review on a Web page for application profiles.

6.4.3. Each Review will include, at a minimum:

6.4.3.1. Any comments from the Usage Board on the application profile.

6.4.3.2. Pointers to locally archived copies of the application profile as originally submitted and (if necessary) as subsequently amended in light of Usage Board comments.

6.4.3.3. A pointer to the "latest version" of an application profile held by its maintainers.

6.5. Review represents a form of recognition, and its URL will be persistent for purposes of citation.

Procedure for approval of DCMI Metadata Terms and Recommendations

Creator: Makx Dekkers
Date Issued: 2004-12-20
Identifier: http://dublincore.org/usage/documents/2004/12/20/approval/
Replaces: http://dublincore.org/usage/documents/2003/08/14/approval/
Is Replaced By: Not Applicable
Latest version: http://dublincore.org/usage/documents/approval/
Status of document This is a DCMI Process Document
Description of document: This document lists the step-by-step process for the approval of DCMI metadata terms and other recommendations.
Recent changes Shepherd of proposals related to metadata terms announces start of public comment period, not Usage Board chair; added link to detailed Usage Board process description.

Procedure for submission of proposals from DCMI Working Groups

Step	Event	Action	Result
1	Open issue or requirement is identified by the community and raised in a Working Group	Working Group (WG) Chair creates an Open Issue Item and adds it to WG Open issue list	Open Issue list change
2	Discussion on Working Group mailing list leads to proposal(s) for solution and identification of responsible authors/editorial team	WG Chair adds the deliverable to the WG Task List, including authors/editorial team responsible for the document and a target delivery date	WG Task list change
3	Authors/editorial team finalize draft document	Authors/editorial team (in consultation with WG Chair) post message to WG and link document from WG pages	DCMI Working Draft
4	WG Discussion	WG Chair manages the discussion and iterative review, and coordinates revision of proposal with the authors/editorial team	DCMI Working Draft (revisions)
5	Consensus reached in WG	WG Chair summarizes consensus and posts last call to WG	DCMI Working Draft (final)
6	Resolution of all comments	WG Chair submits to Managing Director	DCMI WG Proposal
7	DCMI Managing Director receives DCMI WG Proposal	DCMI Managing Director assigns WG Proposal to review team (Usage Board if related to Metadata Term Semantics or (subset of) Advisory Board plus external reviewers otherwise).	Review team established

Continuation of procedure for review and recommendations for technical proposals

Step	Event	Action	Result
8	Review starts	DCMI Managing Director assigns a shepherd to the proposal	DCMI WG Proposal Review in process
9	Review team discussion	Proposal shepherd manages the discussion in the review team and coordinates Review comments	Review Team discussion results
10	Review Results submitted to DCMI Managing Director	DCMI Managing Director and Review Team Chair, in consultation with proposal editor, evaluate Review Team discussion results and decide to either accept (possibly with changes) or reject the proposal	If accepted, proposal becomes a *DCMI Proposed Recommendation* If rejected, proposal is referred back to the WG for further discussion
11	Final text for *DCMI Proposed Recommendation* available	DCMI Managing Director posts DCMI Proposed Recommendation to DC-General and DCMI Web site for Public Comment	Public Comment period commences
12	Public Comment period (minumum four weeks)	Proposal shepherd manages Public Comment period	Public Comments from community
13	Public Comment period finishes	Review team evaluates Proposal and Public Comment results and recommend approval (possibly with changes) or rejection with cause	Recommendation for approval or rejection to DCMI Managing Director
14	DCMI Directorate receives recommendation	DCMI Managing Director, in consultation with WG Chair, review team and others, decides on outcome of process and makes public announcement	If accepted, proposal becomes *DCMI Recommendation* If rejected, special action
15	Final text for *DCMI Recommendation* available	DCMI Managing Director posts *DCMI Recommendation* to DC-General and DCMI Web site	Process completed

Continuation of procedure for review and acceptance of proposals related to metadata terms

For a more detailed description, see the DCMI Usage Board Administrative Processes document.

Step	Event	Action	Result
8	Review starts	Usage Board chair assigns a shepherd to the proposal; proposal is posted on DCMI Web site for Public Comment; shepherd announces to DC-General	Public Comment period commences
9	Public Comment period (minimum four weeks)	Proposal shepherd manages Public Comment period	Public Comments from community
10	Public Comment period finishes	Proposal shepherd prepares for Usage Board discussion	Summary of Public Comments and proposed resolution for Usage Board
11	Usage Board meeting or conference call	Usage Board discusses proposal, taking into account Public Comment results, according to established guidelines and criteria and recommend approval or rejection with cause	Usage Board recommends approval or rejection to DCMI Managing Director

12	DCMI Managing Director receives recommendation	DCMI Managing Director, in consultation with Usage Board Chair and others, decides on outcome of process and makes public announcement	If accepted, proposed term becomes part of *DCMI Metadata Terms* If rejected, proposal is referred back to the WG for further discussion
13	Proposal accepted	Usage Board Chair prepares new version of authoritative documentation; Web team loads new term into http://dublincore.org/documents/dcmi-terms/	Process completed

Metadata associated with this resource: http://dublincore.org/usage/documents/approval/index.shtml.rdf

Title:	**XML, RDF, and DCAPs**
Creator:	Pete Johnston
Date Issued:	2005-02-17
Identifier:	http://www.ukoln.ac.uk/metadata/dcmi/dc-elem-prop/2005-02-17/
Replaces:	Not applicable
Is Replaced By:	Not applicable
Latest Version:	http://www.ukoln.ac.uk/metadata/dcmi/dc-elem-prop/
Description of Document:	This document describes the differences between XML and RDF, and between DC elements, XML elements and RDF properties. It seeks to clarify the requirements that must be met before a "term" can be referenced in a Dublin Core Application Profile (DCAP).

Contents

1. Introduction

For some time DCMI has advocated the approach that the terms of the Dublin Core metadata vocabularies can be deployed in combination with similar terms defined by other sources. This has led to the development of the concept of the "Dublin Core Application Profile" (DCAP) [DCAPUB], as a specification which:

- lists the terms that are used within a class of DC metadata descriptions
- (optionally) describes constraints on how those terms are used within those DC metadata descriptions

A DCAP may be specific to a single application, or it may reflect the usage of an implementer community.

Although the DCMI Usage Board has procedures in place to "review" DCAPs that are submitted to it for evaluation, DCMI does not currently have a formal specification for what constitutes a DCAP. The CEN CWA that described how to present a human-readable representation of a DCAP [CWA14855] described a fairly "permissive" notion of a DCAP. Perhaps as a consequence, implementer interpretations of the concept have tended to vary somewhat; in particular, there appears to be some divergence amongst DCAP designers regarding the nature of the "terms" that are referenced or "used" within a DCAP.

The DCMI Abstract Model [DCMIAM] describes a conceptual framework for Dublin Core metadata descriptions: it describes the logical components which make up DC metadata descriptions and the relationships between them. Although the notion of the DCAP is not explicitly addressed within the Abstract Model, if a DCAP is to specify how a particular set of DC metadata descriptions are constructed, then it follows that the types of "term" referenced within a DCAP must correspond to the types of component described within the Abstract Model.

This document examines some of the specifications used for the representation of data, and particularly the data models used within those specifications. It seeks to clarify some of the terminology and concepts used within those specifications, and in particular to highlight significant differences between concepts that may at first appear to be similar.

It concludes by returning to the question of the DCAP and makes some suggestions on what is required to provide "terms" that *are* usable in DC metadata descriptions, and so are appropriate for reference from a DCAP.

Note: This document provides a good deal of technical background information. It is intended principally for the DCMI Usage Board, rather than as a document for general circulation. Once agreement is reached about the nature of the problem and possible approaches to solving it, then a more concise, user-oriented summary of recommendations for good practice might be produced.

2. XML, XML Elements, XML Namespaces and XML Languages

2.1 XML

The XML 1.0 specification [XML] defines a means of describing structured data in a text-based format. XML uses **tags** embedded in the content of a document to delimit and label parts of the document, and those parts are known as **XML elements**. Tags themselves begin and end with special characters (<....>) so that they can be distinguished from the **element content**, and XML element **end tags** can be distinguished from **start tags** by a special character combination (</...).

The start and end tags include an **XML element type name** and may also contain **XML attributes** (see below). XML elements may contain character data (only), other XML elements, a combination of character data and XML elements - or nothing, i.e. XML elements can be empty. (See Note 1.)

An **XML attribute** is a pair made up of an **attribute name** and an **attribute values**. Multiple XML attributes can occur within the start tag of an element, but each start tag can contain only one XML attribute with a given attribute name. XML attribute values can contain only character data.

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

<metadata>
  <title lang="en">DCMI Home Page</title>
</metadata>
```

Example 1

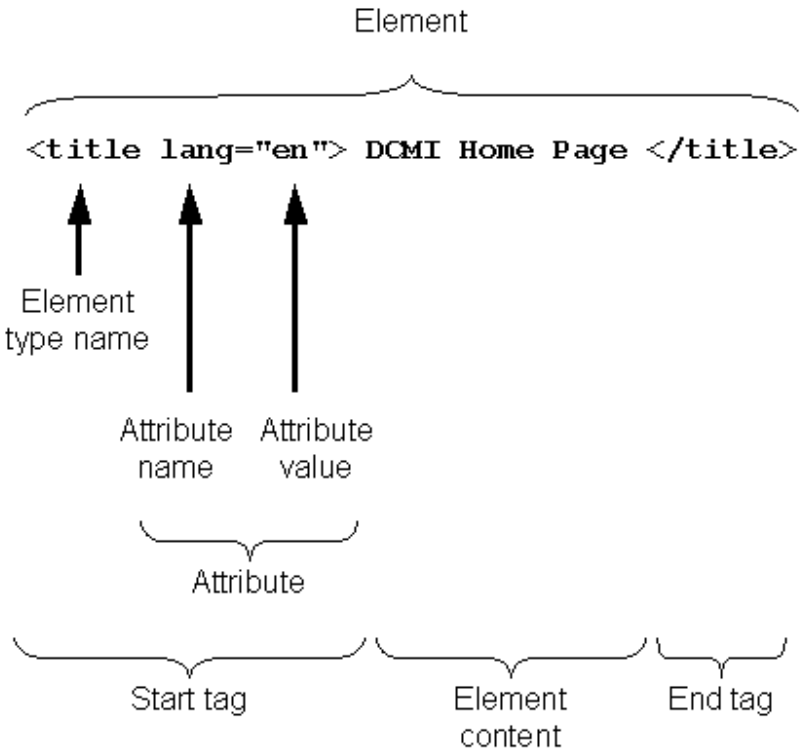


Figure 1: XML Elements and XML Attributes

This document uses the term **component** to refer to XML elements and XML attributes.

XML does not provide a fixed set of element type names and attribute names. Rather users of XML define their own sets of element type names and attribute names for use in tags in XML documents. For this reason, XML is sometimes referred to as a **"meta-language"**, a set of rules for defining XML languages.

2.2 XML DTDs and XML Schemas

XML Document Type Definitions (DTDs) [XML] and XML Schemas [XMLS] provide means of describing/defining constraints on the **structure** of a class of XML documents, the structural relationships that can exist between components: for a named XML element type, the names of the child XML elements it can contain and the XML attributes can be associated with it, and so on. i.e. XML Schemas and XML DTDs describe **content models** for named XML element types and attributes. XML Schema also introduces a datatyping mechanism which is not discussed further in ths document.

An XML document which conforms to the rules of the XML specification and to the structural constraints described by an XML DTD or XML Schema is described as **valid**.

An XML document is described as **well-formed** if it meets certain syntactic constraints: simplifying slightly, well-formedness requires that the document contains only one outermost XML element (the root element), that each XML element has a start and end tag, and that tags are not overlapping. An XML document can be well-formed without being associated with an XML DTD or XML Schema.

2.3 XML Namespaces and XML Qualified Names

As noted above, users of XML define sets of element type names and attribute names for use in tags in XML documents. Further, it can be useful to (re)use independently defined sets of names in combination within the same XML document. However, this raises the prospect of collisions between names which have been defined in multiple name sets.

The *Namespaces in XML* specification [XMLNS] seeks to address the problem of name collisions by providing a mechanism for giving **expanded names** to XML elements and XML attributes. An expanded name is a pair made up of two parts: an **XML Namespace Name** (which is a URI reference) and a **local name**. N.B. An expanded name is *not* itself a URI reference.

Namespaces in XML also introduces the **XML Qualified Name (QName)** as a syntactic construct for deploying expanded names in XML documents. A QName consists of a prefix and a local part. Namespaces are applied to XML elements and XML attributes through the mechanism of a **namespace declaration** which applies to all XML element and XML attribute names within its scope which have a prefix that matches that specified in the declaration. The namespace declaration is said to "bind" a prefix to an XML Namespace Name.

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

<metadata xmlns:dc="http://purl.org/dc/elements/1.1/">
  <dc:title xml:lang="en">DCMI Home Page</dc:title>
</metadata>
```

Example 2

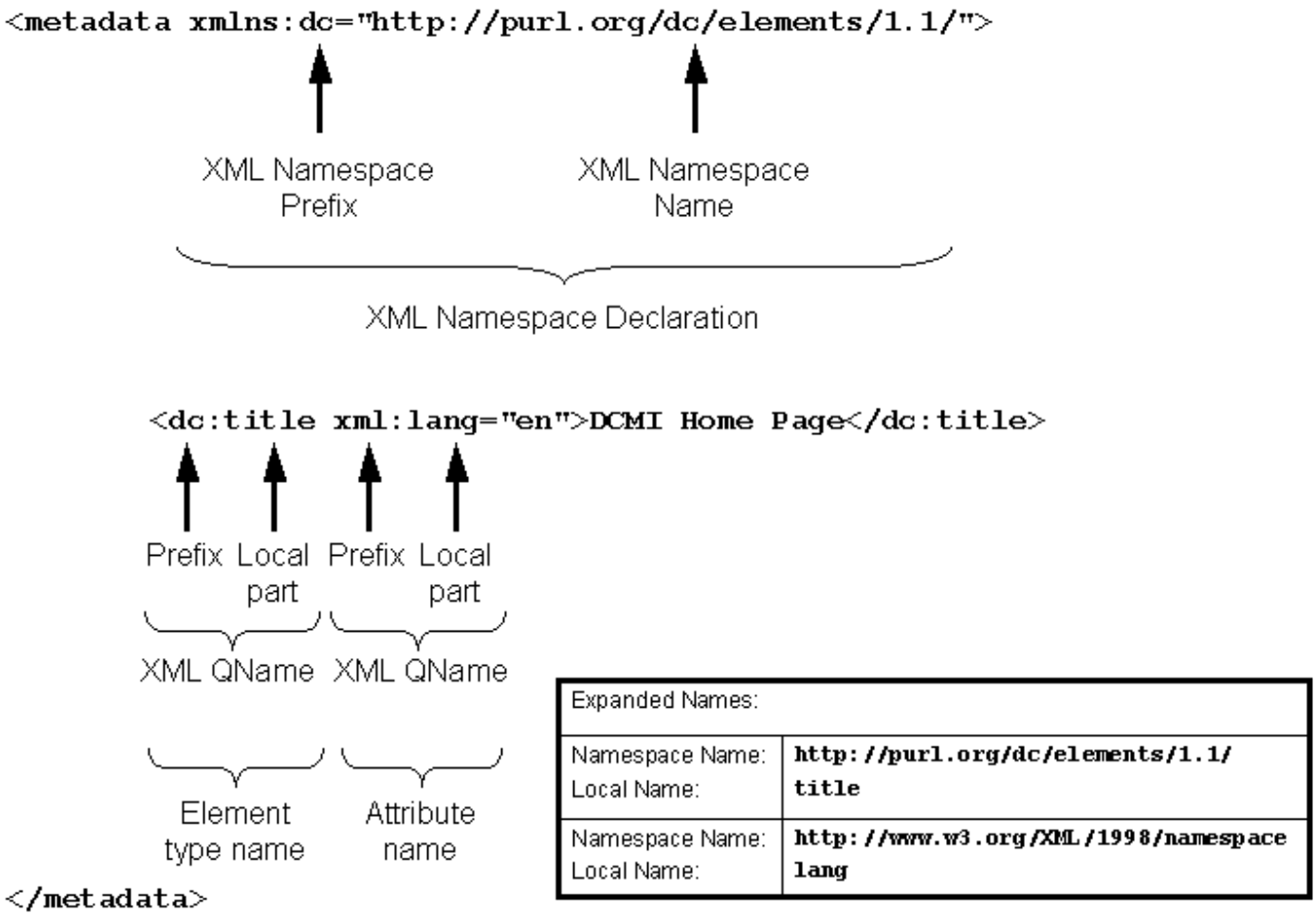


Figure 2: XML Namespaces

(The prefix xml is reserved and does not require an XML Namespace declaration; it is bound to the namespace name `http://www.w3.org/XML/1998/namespace`).

An **XML Namespace** is a collection of names of XML element types and attributes. N.B. It is only a collection of **names**, not a collection of XML elements and attributes. Further, within a single XML document, the same expanded name may be used as both an XML element type name and an XML attribute name (e.g. in an RDF/XML document the expanded name ("http://purl.org/dc/elements/1.1/", "title") may be used, encoded as an XML QName `dc:title`, as both an XML element type name and an XML attribute name.

It is important to note that the XML Namespaces specification provides *only* a means of disambiguating the *names* of components in an XML document: the XML Namespaces specification does *not* provide a basis for "merging" together two XML documents. This is discussed further in the next two sections.

2.4 The XML Infoset: the XML "Abstract Model"

A well-formed XML document can be represented as a **tree structure**, and the XML Information Set [XMLINFO] is an abstract model that describes the set of **information items** of different types which are available from any well-formed XML document. Conversely, any well-formed XML document can be viewed simply as a representation of an XML Information Set.

For example. this XML document

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

<metadata xmlns:dc="http://purl.org/dc/elements/1.1/">
  <dc:title xml:lang="en">DCMI Home Page</dc:title>
  <dc:description xml:lang="en">DCMI is an open forum engaged in the
  development of interoperable online metadata standards.</dc:description>
  <dc:publisher>DCMI</dc:publisher>
  <dc:subject>metadata</dc:subject>
  <dc:subject>resource discovery</dc:subject>
</metadata>
```

Example 3

would be represented as the following tree of XML InfoSet items. (See Note 2)

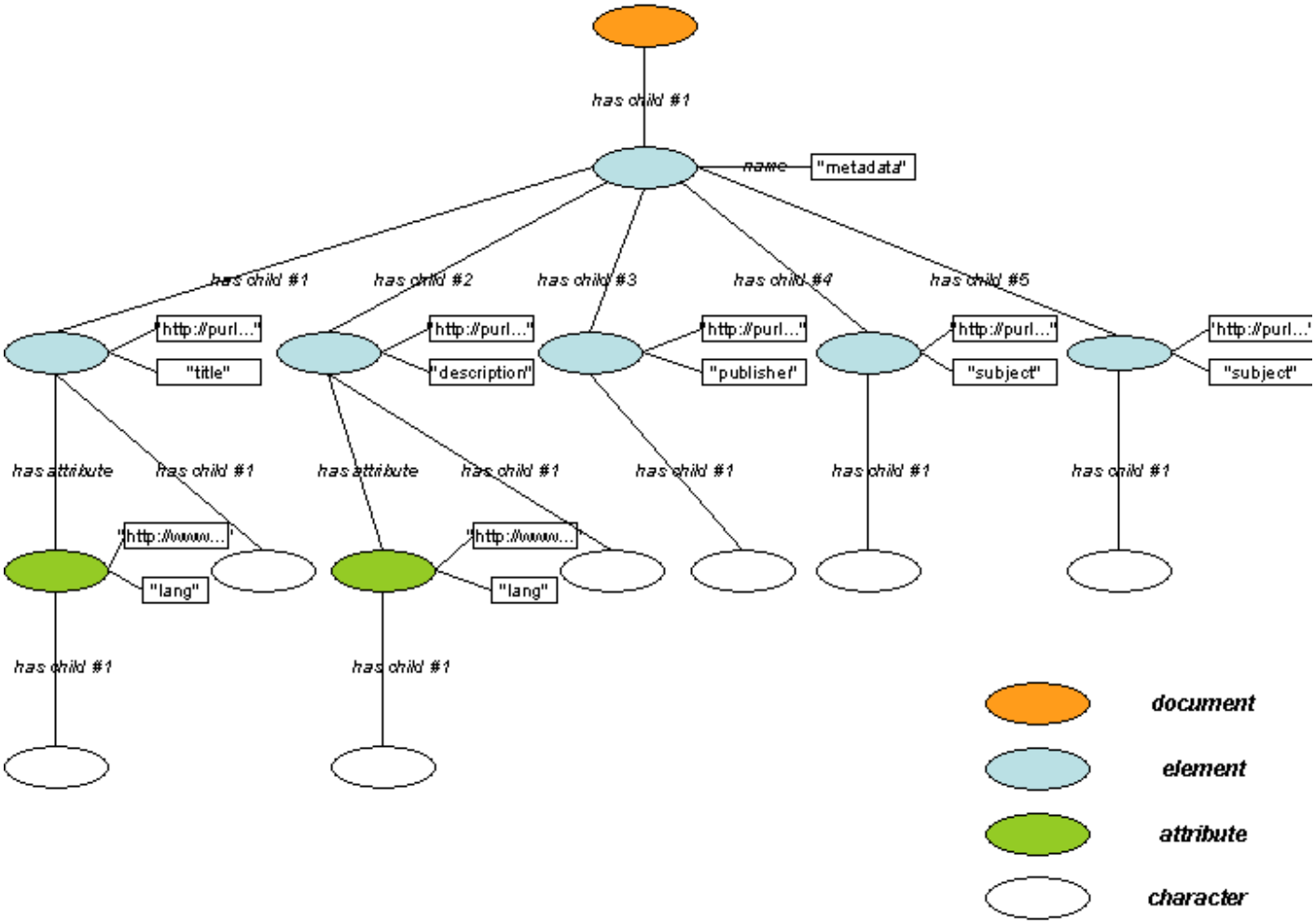


Figure 3: The XML Infoset

Note that:

- the types of relationships between items are defined by the XML Infoset and are (at least for the subset of items considered here) parent-has-child/child-has-parent and element-has-attribute/attribute-is-owned-by-element relationships.
- the InfoSet has one root/document item, and each other item has exactly one parent (for an element) or owner (for an attribute)
- the order of the child information items is represented
- information items are not uniquely identified and several information items are associated with the same expanded name (i.e. the two <dc:subject> elements). Information items can be uniquely addressed, but only by some reference to their context in the tree structure, their relationship to other items

Although the XML Infoset is not a specification for an application programme interface, and XML APIs typically do not present all the information described by the XML Infoset specification, most XML parsers present this type of "view" of an XML document. Similarly specifications like XPath (for addressing parts of an XML document) [XPATH] and XQuery (for querying XML documents) [XQUERY] are based on a tree view of the document.

Further, although the XML Infoset specification was created after the XML specification, it is possible to take the view that any XML document is simply a serialisation of an XML Infoset, and a number of XML-based specifications are defined with reference to the XML Infoset rather than to the XML specification. It may be helpful to try to think in terms of the XML Infoset, or at least of a tree structure, rather than the text syntax, when making comparisons between XML and RDF (see below).

XML itself says nothing about the intended meaning of element type names and attribute names. Furthermore, information is also conveyed by the structural relationships between components within XML documents, such as the parent-child relationships between nested XML elements or element-attribute relationships. XML does not prescribe any fixed meaning for those structural relationships, and in different XML applications, the same structural relationship many carry quite different meanings.

Even within the same application, the same structural relationship may carry a different significance at different contexts within the tree structure.

Those meanings are typically described in human-readable documents which specify how a particular set of named XML elements types and attributes are to be interpreted. (See below on XML languages).

So for example, suppose the designer of an XML application wants to represent the information that the document with the title "Progress Report" was authored by an entity named "John Smith". They might choose any of the following XML structures:

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

<my:metadata xmlns:my="http://example.org/my/">
  <my:title>Progress Report</my:title>
  <my:author>John Smith</my:author>
</my:metadata>
```

Example 4

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

<your:metadata xmlns:="http://example.org/your/"
   your:title="Progress Report"
   your:author="John Smith"/>
```

Example 5

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

<his:metadata xmlns:his="http://example.org/his/">
  <his:general>
    <his:title>Progress Report</his:title>
  </his:general>
  <his:lifecycle>
    <his:author>John Smith</his:author>
  </his:lifecycle>
</his:metadata>
```

Example 6

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

<her:metadata xmlns:her="http://example.org/her/">
  <her:x>
    <her:t>Progress Report</her:t>
  </her:x>
  <her:y her:a="John Smith"/>
</her:metadata>
```

Example 7

All of these are good uses of XML, but they result in very different XML Infosets. It is impossible to interpret what meaning is being conveyed in any of these documents unless the author of the document or the designer of the XML application provides a description which explains what the names of the components and the structural relationships between those components are intended to convey. A human reader (or at least an English-speaking one!) may be tempted to guess at the interpretation based on the names of the components, but as the last example illustrates, XML imposes no requirement that names are drawn from human languages.

Similarly a software application querying these documents would have to be programmed to navigate the four different tree structures. The question "What is the name of the author of the work titled 'Progress Report'?" must be translated into a different query on the tree structure in each case.

2.5 Names in XML Vocabularies and XML Languages

Effective information exchange using XML depends on the sender and receiver of the XML document having a common understanding of the meaning conveyed by the names used in the XML document and by the structural relationships between named components in the XML document. That is, information exchange depends of the on the shared use of **XML languages** (or **formats**) and on the sender and receiver having a common understanding of the rules of the XML language. All XML documents are instances of XML languages, and the interpretation of an XML document is determined by the specification of an XML language.

Such an **XML language** or format has three parts:

1. An **XML vocabulary**: a set of names, drawn from one or more XML Namespaces (or from none), which are used as XML element type names and XML attribute names.
2. A set of **structural constraints** which specifies how the names are to be used as names of components, and describes the permitted content models for those components. The constraints may be expressed by an XML DTD or XML Schema or by some other formalism, or may simply take the form of a narrative description.
3. A **description** of how the named components, and the structural relationships between those components, are to be interpreted to convey information. (see Note 3)

[image to follow]

Figure 4: XML Vocabularies and XML Languages

It is worth exploring some facets of the complex relationships between names, vocabularies, and languages.

2.5.1 XML languages do not require the use of XML Namespaces

Many XML languages do not make use of XML Namespaces in their vocabularies. Examples include Docbook and Encoded Archival Description (EAD). These two XML languages may include components with the same names, but those components have different content models and the meaning conveyed by those components is different (and is described by the human-readable language specifications).

2.5.2 XML Vocabularies and XML Namespaces

There is no simple correspondence between between the set of names used in an XML vocabulary and an XML Namespace. A vocabulary may draw on names that are associated with multiple XML Namespace Names. And the vocabularies of different XML languages may utilise different sets of names associated with the same XML Namespace. e.g. the XHTML 1.0 specification defines three different XML languages: XHTML Transitional, XHTML Strict and XHTML Frameset. Each uses a *different* XML vocabulary but in each case the set of names is associated with the *same* XML Namespace Name `http://www.w3.org/1999/xhtml`.

2.5.3 One name, one component, different constraints

A *single* name may be used as the name of an XML component (XML element, XML attribute) in *multiple* XML languages, and the named component may be associated with a *different* set of structural constraints in each XML language, e.g. the XML vocabulary of the XHTML Transitional language is a superset of the XML vocabulary of the XHTML Strict languages. However in each of those languages the named components are associated with a different set of structural constraints, different content models.

2.5.4 One name, multiple components

Within a single XML language, a *single* name (whether it is qualified by an XML Namespace Name or not) may be associated with *different* types of XML component. The information conveyed by those different components may be different, even if their names are the same.

For example, XHTML uses the name `link` as the name of both an XML element and an XML attribute, but the information conveyed by those two components is quite different.

[This is actually not a good example as the name of the attribute is not namespace qualified so the name of the element is different from the name of the attribute! But the principle holds! I'll try to find a better example.]

2.5.5 One name, one component, different contexts

Within a single XML language, the way individual components are interpreted is conditioned by their structural relationships with other components (containment relations, element/attribute relations etc). So the same name may occur as the name of a component in different contexts in the tree-structure, and it may convey different meaning in those two contexts. For example, in the XML format used to represent instances of the IEEE Learning Object Metadata standard, an XML element with the expanded name "http:// (typically represented by the QName `lom:language`) may occur in three different contexts in the XML tree structure:

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

<lom:lom xmlns:lom="http://ltsc.ieee.org/xsd/LOM">
  <lom:general>
    <lom:language>en</lom:language>
  </lom:general>
  <lom:metametadata>
    <lom:language>en</lom:language>
  </lom:metametadata>
  <lom:educational>
    <lom:language>en</lom:language>
  </lom:educational>
</lom:LOM>
```

Example 8

The same XML element conveys three different pieces of information depending on its context in the tree structure:

- as a child element of the `lom:general` XML element, it is used to represent the language used within the learning object
- as a child element of the `lom:metametadata` XML element, it is used to represent the language of the metadata instance
- as a child element of the `lom:education` XML element, it is used to represent the language of a typical user of the learning object

2.5.6 Ordering of components

A variant of the previous case of context conditioning interpretation is that the ordering of components may be significant in an XML language. In the LOM XML binding, ordering is considered significant in several parts of the tree-structure e.g. a sequence of source/value XML element pairs is used to represent a list of learning resource types, and according to the LOM standard, "The most dominant kind shall be first".

2.5.7 One name, different components, different languages

Consider the following three documents:

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

<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
            xmlns:dc="http://purl.org/dc/elements/1.1/">
  <rdf:Description dc:title="DCMI Home Page" />
</rdf:RDF>
```

Example 9: RDF/XML

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

<description xmlns:dc="http://purl.org/dc/elements/1.1/">
  <dc:title>DCMI Home Page</dc:title>
</description>
```

Example 10: DC-XML

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

<xsl:stylesheet version="1.0"
                  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
                  xmlns:dc="http://purl.org/dc/elements/1.1/">

  <xsl:template match="/"">
    <dc:title>
      <xsl:text>DCMI Home Page</xsl:text>
    </dc:title>
  </xsl:template>
</xsl:stylesheet>
```

Example 11: XSLT

The XML QName `dc:title` occurs as the name of a component in all three XML documents, and in each case it corresponds to the same expanded name ("http://purl.org/dc/elements/1.1/", "title").

In Example 9, it is the name of an XML attribute, and when the XML document is interpreted following the rules of the RDF/XML specification [RDFXML], the XML Infoset is interpreted as representing a single RDF triple.

In Example 10, it is the name of an XML element, with text only content, and if the XML document is interpreted following the rules of the DC-XML language described in *Guidelines for Expressing DC in XML* [DCXML] [assuming they were better written!], the document is interpreted as representing a Dublin Core metadata description, consisting of a single statement about an unidentified resource.

In Example 11, it is again the name of an XML element, but this time with a single child XML element. The document is interpreted following the rules of the XSLT XML language [XSLT], and the expanded name ("http://purl.org/dc/elements/1.1/", "title") is interpreted not as part of a DC metadata description, but simply as the name of an element node to be added to the XML result tree generated by the XSLT transformation.

2.5.8 Summary

The aim of providing these detailed examples is to illustrate that the same XML expanded name (XML Namespace Name, Local Name pair) may be used in **multiple** XML vocabularies and in **multiple** XML languages. While the XML Namespaces specification provides for the avoidance of name collisions, it does not address the question of what it means to mix named components from different XML languages. XML components (XML elements and XML attributes) are not, in the general case, "stand-alone" and can not be interpreted independently of their context in an XML document. Meaning in XML documents is derived from combinations of named components. There is some expectation that the name has a consistent meaning across XML languages, but the meaning of a named component is always **scoped by the XML language in which it occurs**, and even within a single XML language, the meaning of a named component may be dependent on **the context of that component within the tree structure**.

2.6 Modularity and Extensibility in XML Languages

The ability to use components of an XML language independently of other components of the language and to (re)use components that are specified within one XML language in the context of another XML language should not be taken for granted. If an XML language is to be extensible, that extensibility must be built into the design of the language.

Some XML languages are defined as essentially standalone and are intended to be used more or less by themselves (e.g. TEI, XHTML). But some XML languages are created to be used in association with other languages.

Some are container languages where the expectation is that they will act as a wrapper for components (sometimes referreed to as a "payload") which are themselves constructed according to the rules of another language, where this second language may not even be known at the time the container language is designed. Examples of container languages include the SOAP or OAI-PMH formats. The containment function is defined within the rules of the SOAP language: a receiver of a SOAP XML instance interprets that document according to those containment rules, but the contained component is interpreted according to the rules of a second language. Similarly METS, although not only a container language, has well-defined components which do act as containers for other XML formats

At the other extreme are XML languages are intended for use within the context of other languages. For example, MathML can be used stand-alone, but is also intended to be embedded within other languages. Some XML languages can *only* be deployed in the context of another language e.g. languages like XLink or RDF/A provide only XML attributes, which are intended for use on the XML elements defined by another XML language. (Such languages are sometimes referred to as "parasite" languages as they require a "host".)

[Middle-ground examples e.g. RSS2.0/Atom, RDF/XML, DC in XML - they work because there is another data model layered on top of the XML InfoSet]

2.7 Summary

- XML QNames (as representations of expanded names) are associated with components of XML documents (XML elements, XML attributes). Multiple XML elements in the same document may have the same expanded name.
- XML DTDs and XML Schemas describe structural constraints on a class of XML documents
- An XML language consists of an XML vocabulary (a set of names), a set of structural constraints (which may be expressed by a DTD or XML Schema), and a human-readable specification of how the named components are interpreted
- XML imposes no fixed meaning on the structural relationships between XML components (parent-child relationships between elements or owner relationships between attributes and elements), and the same relationship carries different meaning in different XML languages and in different contexts within the same XML language

- The same XML QName (representing an expanded name) may be used in the XML vocabulary of multiple XML languages; in each of those XML languages, the named component may be subject to different structural constraints
- Although there is some expectation that names are used consistently across XML languages, the meaning of a name is dependent on the XML language in which it occurs
- The interpretation of an XML document depends not only on the XML vocabulary, but on the rules of the XML language, and these rules - the "semantics" of the language - are available only to the human reader. In particular, they are not part of the XML Infoset, the XML "abstract model".

3. RDF, URI references, and RDF/XML

3.1 The RDF data model and URI references

The Resource Description Framework (RDF) set of specifications describe a means of constructing simple statements about resources.

Central to RDF is the idea of the resource, which can be anything you wish to describe - a document, a physical object, a person, an imaginary being, a concept, anything at all - and the idea of identifying resources using Uniform Resource Identifiers (URIs) (or more accurately URI references). In RDF, URI references are simply names for things. The fact that some URI references used in RDF may also be used by software applications to obtain access to digital objects is irrelevant to RDF. Also RDF treats URI references as "opaque" strings: the internal structure of a URI reference has no significance in RDF. It is important to note that an RDF application can not determine the relationship between a URI reference and a resource - it can only make use of the URI reference as a name.

(The nature of the relationship between URI references and resources has been part of the debate about "social meaning" in RDF. Essentially, URI references are used as if they always identify/denote a single resource, but that assumption is not part of the formal semantics of RDF. (I think I've got that right, but I may be oversimplifying.))

The basic building block of the RDF data model is the **triple**, consisting of a subject, a predicate and an object. The **subject** is a URI reference (or a "blank node"), the **predicate** is a URI reference, and the **object** is a URI reference, a blank node or a literal. (This document will not deal with "blank nodes" in any detail - for the purposes of the current discussion a blank node can be considered to be a sort of local identifier for a resource which is not identified by a URI reference.)

Each triple represents a statement: that statement asserts that a relationship exists between the two resources denoted by the subject and the object of the triple, and the type of that relationship is indicated by the predicate URI reference. A URI reference that is used as the predicate of a triple denotes a particular type of resource called a **property**.

As noted above, RDF does not deal with the relationship between a URI reference and the resource it denotes. Although this level of "meaning" - the difference between "having a title" and "having a subject", for example - may be used by the human interpreters of RDF statements, or by programmers writing software to operate on RDF data - it is not accessible to software. However, the RDF specifications, specifically *RDF Semantics* [RDFSEM], do provide a "formal meaning" for RDF and for the sets of URI references (vocabularies) defined by the RDF specifications. This "formal meaning" is defined in terms of the logical inferences that can be drawn, the "entailments" that follow, from the use of those URI references in RDF statements.

The following four triples represent four statements, each one stating a relationship between two resources:

Subject	Predicate	Object
http://example.org/doc/123	http://purl.org/dc/elements/1.1/creator	http://example.org/person/John
http://example.org/doc/456	http://purl.org/dc/elements/1.1/contributor	http://example.org/person/John
http://example.org/person/John	http://xmlns.com/foaf/0.1/name	"John Smith"
http://example.org/person/John	http://xmlns.com/foaf/0.1/knows	http://example.org/person/James

Example 12

Since RDF triples by definition accommodate only one subject and one object, a property describes a relationship between two resources, a binary relation. So a property is a "conceptual resource". It is still a resource, however, and a property URI reference can be the subject or object of an RDF triple, i.e. RDF allows you to create statements "about" a property in the same way as about other types of resource.

Subject	Predicate	Object
http://purl.org/dc/elements/1.1/creator	http://www.w3.org/2000/01/rdf-schema#label	"Creator"
http://purl.org/dc/elements/1.1/creator	http://www.w3.org/2000/01/rdf-schema#comment	"An entity primarily responsible for making the content of the resource."

Example 13

While the abstract model of an XML document is a tree, the abstract model for RDF is a "graph": a structure where "nodes" are linked together by "arcs". The subject and object of a triple are represented by nodes and the predicate is a labelled arc linking from the subject node to the object node. The triples in Example 12 would be represented as the following graph:

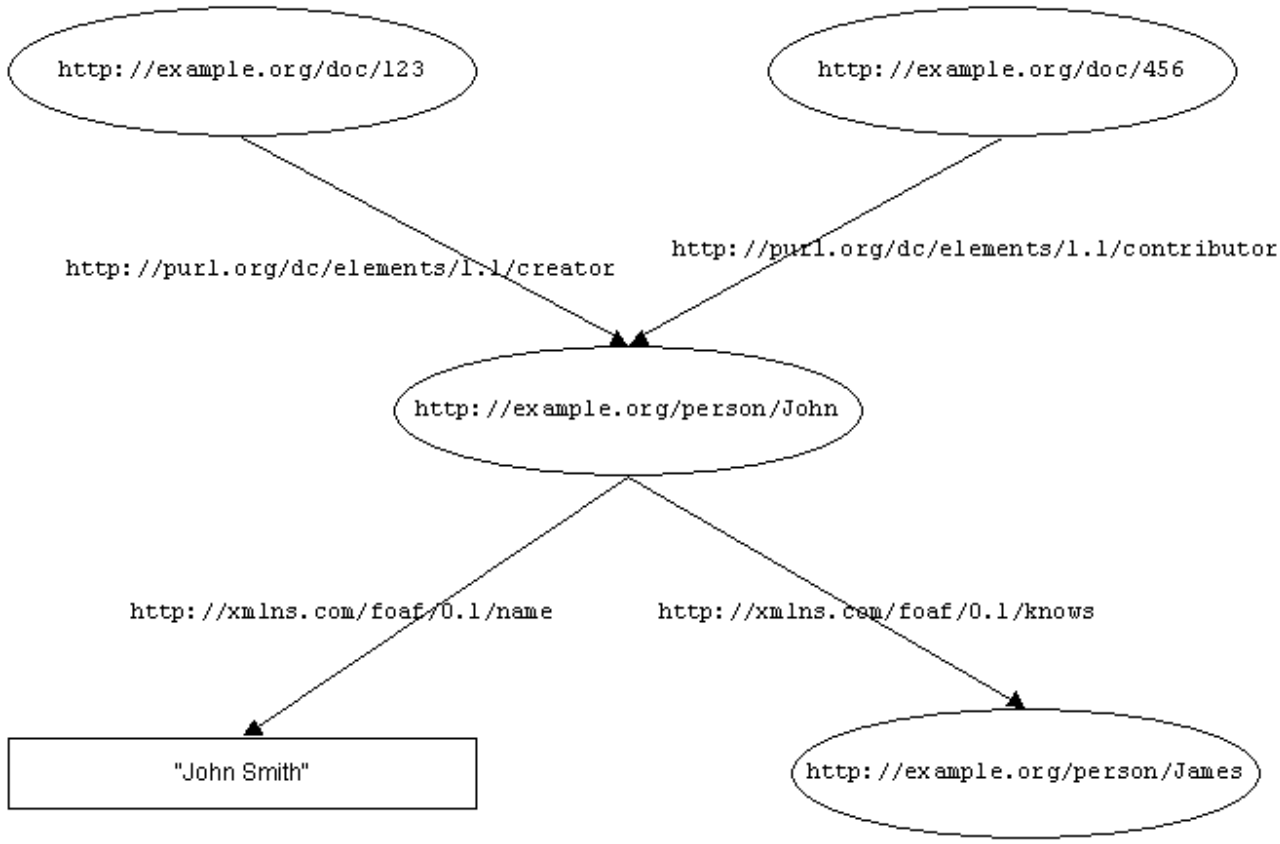


Figure 5: The RDF Graph

Just as the XML Infoset tree is an alternative view of an XML document, so the RDF graph is an alternative view of the subject-predicate-object triples.

In the RDF graph, the nodes are URI references that name resources of any type, and any node may be linked to an unlimited number of other nodes, and each of those links may carry any URI reference as a label. There is no order in an RDF graph.

The key difference between the XML Infoset tree and the RDF graph is that the RDF data model specifies that each node-arc-node triple is to be interpreted as a set of statement, whereas XML leaves it to each separate XML language specification to describe how the parent-child and attribute-element relationships in the tree are to be interpreted.

The triple/graph model also makes it easy to merge together two different graphs, two different sets of triples. The merged graph is simply the "union" of the two individual graphs, or the concatenation of the sets of triples, *but* with care taken to ensure that blank nodes (local identifiers) are maintained as distinct. This means that combining data from different sources, which is complex using XML, is relatively easy using RDF.

3.2 RDF Vocabularies

In the same way that XML does not provide a fixed set of XML element type names and attribute names, so RDF does not specify a fixed set of URI references that can be used in RDF triples. Rather RDF user communities deploy URI references that denote resources of interest to them. They need not only URI references to denote the particular resources (documents, books, images, concepts etc) they wish to describe, but also URI references to indicate the types of those resources and the properties used to describe their attributes and the relationships between them i.e. user communities define RDF **vocabularies** for their domains of interest.

The RDF Vocabulary Description Language (RDF Schema) [RDFS] provides....

(Something about classes and type-ing, subproperty/subclass)

(N.B. URIref opacity - tells you nothing about vocabulary etc.)

3.3 Syntaxes for serialising RDF

In order to exchange RDF data between applications, the data must be represented in some digital format. This process is referred to as **serialisation**. The RDF data model is independent of any specific serialisation syntax. In particular RDF does not rely on XML. There are several XML-based syntaxes for representing sets of RDF statements, and there are also several syntaxes for that are not based on XML.

3.3.1 RDF/XML

The RDF/XML specification [RDFXML] provides a set of rules for representing a set of RDF triples in XML. In the terms of the discussion of XML above, the RDF/XML

specification defines RDF/XML as an XML language.

The RDF/XML language specification defines a convention for representing RDF URI references as expanded names, encoded in documents as XML QNames. It is important to remember that there is a mapping taking place between XML QNames in the XML document and RDF URI references in the RDF graph, and that this is a convention specific to the RDF/XML language. It is *not* the case that the XML expanded name or the XML QName identifies the RDF property. And indeed a single URI reference may be expressed in RDF/XML using many different XML QNames. See Example 14 and Example 15 below.

Further, RDF/XML represents only some URI references as XML QNames (predicate URI references and URI references that represent the type of a resource: other URI references are encoded in full. (Also of course there are XML QNames used in RDF/XML that name components of the RDF/XML language but do not map to URI references (e.g. `rdf:Description`, `rdf:resource`, `rdf:parseType` etc).

The triples in Example 12 could be represented in RDF/XML as follows. All of these XML documents are alternate representations/serialisatiions of the *same* RDF graph.

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

<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
            xmlns:dc="http://purl.org/dc/elements/1.1/"
            xmlns:foaf="http://xmlns.com/foaf/0.1/">

  <rdf:Description rdf:about="http://example.org/doc/123">
    <dc:creator rdf:resource="http://example.org/person/John"/>
  </rdf:Description>

  <rdf:Description rdf:about="http://example.org/doc/456">
    <dc:contributor rdf:resource="http://example.org/person/John"/>
  </rdf:Description>

  <rdf:Description rdf:about="http://example.org/person/John">
    <foaf:name>John Smith</foaf:name>
    <foaf:knows rdf:resource="http://example.org/person/James"/>
  </rdf:Description>


</rdf:RDF>
```

Example 14

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

<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
            xmlns:dc="http://purl.org/dc/elements/1.1/"
            xmlns:foaf="http://xmlns.com/foaf/0.1/">

  <rdf:Description rdf:about="http://example.org/doc/123">
    <dc:creator>
      <rdf:Description rdf:about="http://example.org/person/John" foaf:name="John Smith">
        <foaf:knows rdf:resource="http://example.org/person/James"/>
      </rdf:Description>
    </dc:creator>
  </rdf:Description>

  <rdf:Description rdf:about="http://example.org/doc/456">
    <dc:contributor rdf:resource="http://example.org/person/John"/>
  </rdf:Description>

</rdf:RDF>
```

Example 15

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

<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
            xmlns:z="http://purl.org/dc/elements/1.1/c"
            xmlns:foaf="http://xmlns.com/foaf/0.1/">


  <rdf:Description rdf:about="http://example.org/doc/123">
    <z:reator>
      <rdf:Description rdf:about="http://example.org/person/John" foaf:name="John Smith">
        <foaf:knows rdf:resource="http://example.org/person/James"/>
      </rdf:Description>
    </z:reator>
  </rdf:Description>

  <rdf:Description rdf:about="http://example.org/doc/456">
    <z:ontributor rdf:resource="http://example.org/person/John"/>
  </rdf:Description>

</rdf:RDF>
```

Example 16

Example 16 was designed to highlight that RDF/XML makes a mapping between XML expanded names and URI references. Example 15 and Example 16 have the same

XML document structure, a corresponding set of XML components. But those components have a different set of XML expanded names in each case. Yet they both represent the same RDF graph, the same set of RDF triples. It is important to remember that, to an RDF application, these variations in the serialisation syntax - which *would* be significant in an XML application - are quite invisible and have no significance: the QNames `dc:creator` (in Example 14 and Example 15) and `z:reator` (in Example 16) are both simply a means of representing the single URI reference `http://purl.org/dc/elements/1.1/creator`, and many other prefix/namespace name/local name permutations are possible.

3.3.2 Other syntaxes

RDF/XML is just one syntax for the serialisation of RDF graphs. There are other XML-based syntaxes (e.g. TRiX,) and also text-based syntaxes not based on XML (e.g. N-Triples, Turtle etc). Many of these syntaxes incorporate a mechanism which permits the encoding of URI references using Qualified Names, though in the case of the non-XML syntaxes these conventions are unrelated to the concept of the **XML** Namespace. A single RDF application might read and write documents in many different RDF serialisation syntaxes, but all the different formats are representations of graphs, sets of triples.

For these reasons, it is important when discussing RDF - and particularly when comparing RDF and XML - to try to focus on the "abstract models" of the RDF graph and the XML tree. Comparison at the syntactic level may lead to confusion and false conclusions, particularly (as the examples above show) regarding the significance or otherwise of the *names* (QNames, expanded names) used to label XML components. This can be difficult at first for people accustomed to reading XML documents, but it is an absolutely vital step.

3.4 Summary

- URI references are identifiers for resources of any type.
- RDF provides a simple data model for making simple statements about resources in the form of triples composed of a subject, predicate and object. RDF statements make use of URI references as names.
- The RDF "abstract model" is that of a graph, with nodes linked together by labelled arcs.
- The RDF data model is defined quite independently of XML, but sets of RDF triples can be serialised as XML documents.
- (RDF Schema)

4. XML and RDF

4.1 Qualified Names in XML amd RDF

In XML, XML QNames are used in XML documents to represent the XML expanded names (two part constructs made up of an XML Namespace Name and a local name) that form the vocabulary of an XML language. Those expanded names are used as the names of components in XML documents (XML elements, XML attributes). They are processed and interpreted according to the specification of that XML language. It must be emphasised that in XML generally, XML QNames are *not* URI references and they are not mapped to URI references.

In RDF, some text-based serialisation syntaxes provide a mechanism for using "qualified names" to abbreviate URI references. And in discussions of RDF generally, it is commonplace to find "qualified names" used, as abbreviations for those URI references, to refer to those properties and classes. So, for example, the property with the URI reference `http://purl.org/dc/elements/1.1/title` is sometimes referred to as `dc:title` or `DC.title`. The qualified name form is simply an abbreviation for the full URI reference.

In RDF/XML, URI references may be represented as XML expanded names, which are encoded as XML QNames used as XML element type names or XML attribute names. However, it is important to bear in mind that the XML components in XML documents are different things from the property itself, and that there is a mapping process taking place which is specific to this XML language.

A focus on the RDF/XML syntax to the exclusion of the RDF data model can lead to false assumptions about the use of names in XML languages and in RDF.

The vocabulary of an XML language (the set of expanded names which is encoded as QNames) i snot the same thing as an RDF vocabulary (a set of URI references). And the existence of an XML vocabulary and the use of the corresponding QNames in XML documents does *not* result in the creation of a corresponding set of URI references. Approaching RDF on the basis that the QNames that have been used in an XML language can simply be redeployed in RDF/XML is not a coherent approach because it ignores the fact that in the two contexts the names apply to quite different entities and are interpreted in quite different ways.

Certainly, an XML QName currently used in any XML language (XHTML, MODS, METS etc,) *could* be deployed in an RDF/XML document. URI references do not have to be pre-declared before they appear in RDF triples. And depending on the context in which that XML QName is used in the RDF/XML document, an RDF/XML parser would generate a URI reference from the expanded name and present that URI reference in an RDF triple. It may appear as the predicate of an RDF triple, and on that basis an RDF application will infer that the generated URI reference denotes a property. But that property is not the same thing as the initial XML component

Consider a concrete example. The XHTML XML language includes an XML element type name `title` associated with the XML Namespace Name `http://www.w3.org/1999/xhtml`. That name can be deployed in an RDF/XML document:

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

<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
            xmlns:xhtml="http://www.w3.org/1999/xhtml">
  <rdf:Description xhtml:title="DCMI Home Page" />
</rdf:RDF>
```

Example 17: RDF/XML

and an RDF/XML parser will generate the triple

Subject	Predicate	Object
(blank node)	http://www.w3.org/1999/xhtmltitle	"DCMI Home Page"

Example 18

The RDF/XML parser generates a URI reference and infers that the URI reference denotes a property, but the XHTML specification does not provide any information

about a resource with the URI reference `http://www.w3.org/1999/xhtmltitle` and there is no RDF Schema description of this property. The XHTML specification describes an XML language and describes only XML components with expanded names, to be interpreted in the context of an XML tree structure. The `xhtml:title` element is described as a container for a text string, not as a property.

Further, using the same XML name as the name of a different component in RDF/XML generates a *different* RDF graph:

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

<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
           xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
           xmlns:xhtml="http://www.w3.org/1999/xhtml">
  <xhtml:title rdfs:label="DCMI Home Page" />
</rdf:RDF>
```

Example 19: RDF/XML

Subject	Predicate	Object
(blank node)	http://www.w3.org/1999/02/22-rdf-syntax-ns#	http://www.w3.org/1999/xhtmltitle
(blank node)	http://www.w3.org/2000/01/rdf-schema#label	"DCMI Home Page"

Example 20

Here the rules of the RDF/XML language dictate that `xhtml:title` is interpreted as representing a URI reference that provides the type of the resource, and an RDF processor infers that that URI reference denotes a class. Again no such interpretation is covered by the XHTML specification.

And consider the rest of the vocabulary of the XHTML language. Any name from that vocabulary (`xhtml:html, xhtml:p, xhtml:em`) could be deployed in the same way, and an RDF triple generated, but such triples do not form - or do not necessarily form - a coherent representation of the information conveyed by the XHTML XML language.

Such an approach is simply transposing names from one context to another with no consideration for the contexts within which the names are deployed and interpreted. In short the names used in an XML language can not simply be transposed into RDF (or rather into RDF/XML), or at least not in any meaningful way. (See Note 4)

To map between XML and RDF - or rather, as discussed in the next section, between an XML language and the RDF data model - , it is necessary to consider not simply the vocabulary of the XML language but the meaning that the XML language is intended to convey, the semantics of the language that are not accessible from the XML Infoset.

Note that it may emerge that that "semantic" analysis/re-modelling/mapping process *does* lead to a decision to use URI references that are encoded using QName forms that are similar to those used in the XML language e.g. in the example above the mapping might specify that an RDFS class called `http://www.w3.org/1999/xhtmltitle` is required. But that decision would then be the result of the considered analysis and re-modelling, taking into account the contexts of the XML language and the RDF data model, rather than a "blind" transfer of the names.

4.2 Mapping XML Languages to the RDF data model

It may well be the case that an XML document *does* represent simple statements about resources. But there is nothing in the XML specification that describes how to represent such statements in the XML tree structure. Different XML language designers make different decisions about how the document author should encode this information in an XML document. As a result, the recipient of the XML document needs access to the specification of the XML language if they are to interpret the XML document as anything other than a simple tree structure.

As a consequence, there is no single way of interpreting an XML tree structure in terms of the RDF data model. Rather it is necessary to:

- analyse the specification of the individual XML language to determine what information is being represented by that language, the semantics of the XML language that are not directly accessible from the XML Infoset
- develop a specification of how that information shoyuld be represented using the RDF data model (using URI references from existing RDF vocabularies, or by developing new RDF vocabularies for the concepts specific to the application domain within which the XML language is used, or by a combination of the two approaches)
- describe the mapping or correspondence between constructs used in the XML language and statements made using the URI references selected or created

This is a re-modelling and mapping process: the names and components used in XML documents are quite different from those used in RDF graphs. It is also an XML-language-specific process because, as described above, the interpretation of names and components varies across XML languages. It may also vary according to the context of the named component within the same XML language. Because of the nature of, and the differences between, the XML and RDF data models, there may be no simple one-to-one correspondence between XML element type names and XML attribute names on the one hand and RDF URI references on the other.

Depending on the design of the XML language, there may be regular "patterns" used in that language which make this mapping process easier, and encouraging the adoption of such patterns may facilitate the development of the mapping. But in the general case there is no one set of rules that can be applied to all existing XML languages. This was summarised concisely in a recent message to the W3C RDF Interest Group mailing list

> There is no default mapping of XML document instances to RDF triples, other than the representation of the infoset in RDF, since XML is a generic framework that allows people to create an unbounded amount of applications on top of it.
>
> - Sean B. Palmer, 2005-01-15

Note: None of the above is intended to suggest that RDF is better or worse than XML, simply that they are different and those differences must not be ignored. Both XML and RDF have their uses, and indeed there are many cases where XML may be a better choice than RDF (e.g. when data deals with N-ary relations: while it may be possible to re-model it as a set of binary relations, that re-modelling may not be efficient.)

5. Dublin Core and Dublin Core Application Profiles

5.1 The DCMI Abstract Model

The DCMI Abstract Model defines a DC metadata **description** as a set of **statements** about a single subject **resource**. Each statement is made up of:

- a reference to a **property**, in the form of a URI reference (a **property URI reference**)
- a reference to a second resource, the **value**, in the form of either a URI reference (a value URI reference), a **representation** of the value or a **description** of the value

A statement may also contain a reference to a **vocabulary encoding scheme** and a **syntax encoding scheme**, again both in the form of URI references. DC metadata descriptions are typically grouped together as **description sets**.

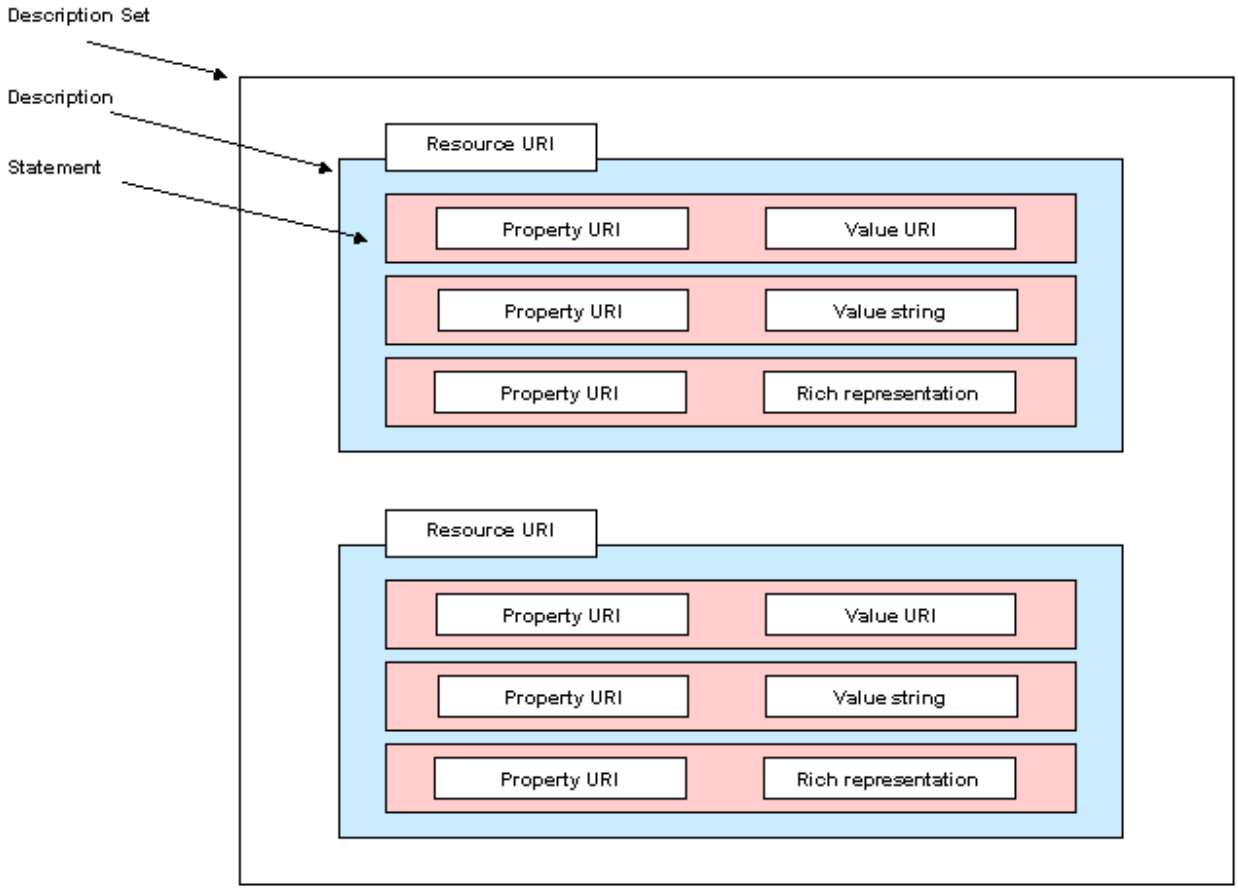


Figure 6: A DC Metadata Description Set

Properties and encoding schemes are referred to in DC metadata descriptions by means of URI references. Without a URI reference, a property or encoding scheme can not be referred to in a DC metadata description.

The Abstract Model is essentially a variation on the RDF data model. Although DC statements appear to have two parts, they are always associated with a description, and a description applies to exactly one resource, so DC statements are really triples.

The Abstract Model differs in

- its treatment of literals
- the introduction of description and description set

But essentially all the comments made above about RDF - and particularly the distinctions between RDF and XML - apply to the DC Abstract Model. The Abstract Model is not based on the XML tree model.

5.2 Dublin Core Terms, "Elements", Qualified Names and URI references

DCMI assigns URI references to all the "terms" it defines following the policies and conventions described in the *Namespace Policy for DCMI Terms* document [DCMINS]. Those URI references take the form of PURLs, e.g. `http://purl.org/dc/elements/1.1/title`.

Note that DCMI uses the word "term" to refer to the conceptual resource rather than the URI reference defined to it.

The "terms" defined by DCMI are of three types:

- elements and element refinements
- encoding schemes
- "terms from a controlled vocabulary"

The nature of these "terms" is described by the DCMI Abstract Model. A DC element or element refinement is a property: "a specific aspect, characteristic, attribute, or relation" that can be applied to the description of a resource. An encoding scheme is a class. Although the Abstract Model distinguishes between vocabulary encoding schemes and syntax encoding schemes, DCMI currently models all encoding schemes as classes.

Although DCMI documentation refers to "terms from a controlled vocabulary", the only controlled vocabulary that it currently maintains is the DCMI Type Vocabulary, and the "terms" in this vocabulary have the specific characteristic that they are all classes. This would not necessarily be the case for other "controlled vocabularies", where the "terms" may be resources of any type.

A DC element is **not** the same type of thing as an XML element (or element type):

- an **XML element** is a component within an XML document and its characteristics are defined by the XML and XML Infoset specifications. XML elements have element content, child elements, and attributes; an XML element has an XML element type name, which takes the form of an expanded name; and the interpretation of an XML element depends on its use in the context of an XML language, which is described by the human-readable specification of that language
- a **DC element** is a property and its characteristics are defined by the RDF specifications and the DCMI Abstract Model; a property is identified by a URI reference, and that URI reference can be used as a predicate in an RDF triple or a property URI in a DC metadata statement

Although all the DC "terms" - properties and classes - are identified by URI references, it is common in DCMI documentation to find "qualified names" used, as abbreviations for those URI references, to refer to those properties and classes. So, for example, the property with the URI reference `http://purl.org/dc/elements/1.1/title` is sometimes referred to as `dc:title` or `DC.title`. Some text-based syntaxes for serialising RDF graphs also support this construction. The qualified name form is simply an abbreviation for the full URI reference.

However, DC metadata descriptions may also be serialised as XML documents, either using the RDF/XML language or the DC-XML language. In both those XML languages, URI references may be represented as XML expanded names, which are encoded as XML QNames used as XML element type names or XML attribute names. However, it is important to bear in mind that the XML components in XML documents are different things from the property itself, and that there is a mapping process taking place which is specific to these XML languages.

Usually, it is possible to establish from the context whether a qualified name is being used as a shorthand for a URI reference or as an encoding of an XML expanded name, but care needs to be taken.

Taken together, however, these two factors - the unqualified use of the word "element" to refer to two different things and the use of qualified names in two different contexts - have contributed to some confusion. It must be emphasised that in XML generally, XML QNames are *not* URI references and they are not mapped to URI references: an XML QName is used to encode an expanded name in an XML document, and an expanded name is a two part construct, made up of an XML Namespace Name and a local name. It is interpreted in the context of the XML language in which it is used. (See Note 5)

Because the DCMI Abstract Model is based on or similar to the RDF data model, all the points made about XML and RDF in section 4 above apply to XML languages and DC metadata applications. The names and components used in an XML language can not be deployed in a DC metadata description: rather it s necessary to follow the process of analysing the meaning that the XML language (or some subset of constructs within the XML language) is intended to convey and developing the *new* set of "terms" required - and that set of "terms" may include properties, classes, and other resources depending on what information is to be represented.

5.3 Dublin Core Application Profiles and XML Languages

As noted in the introduction, DCMI has not defined a formal model for what a Dublin Core Application Profile (DCAP) actually is (See Note 6). Probably the closest to such a model that DCMI has at the present is the statement in the CEN CWA 14855 that:

> A Dublin Core Application Profile (DCAP) is a declaration specifying which metadata terms an organization, information provider, or user community uses in its metadata. By definition, a DCAP identifies the source of metadata terms used - whether they have been defined in formally maintained standards such as Dublin Core, in less formally defined element sets and vocabularies, or by the creator of the DCAP itself for local use in an application. Optionally, a DCAP may provide additional documentation on how the terms are constrained, encoded, or interpreted for application-specific purposes.

However CEN CWA 14855 is not clear about many aspects of a DCAP. In particular, the suggestion that "terms" may be referred to within a DCAP even if they are not identified by URI references has caused confusion.

With reference to the the Abstract Model, it seems reasonable to consider that a DCAP specifies the "terms" that are referenced within a particular class of descriptions or description sets. As discussed in section 5.1 these "terms" are **properties** and **classes**. So the "terms" referenced or "used" in a DCAP are also properties and classes. A DCAP specifes the properties that are used to describe particular types of resource (classes), and how those properties are deployed, including any constraints on their values i.e, any classes to be used as encoding schemes. The properties and classes are referenced by citing their URI references, which may be drawn from any RDF vocabularies, including vocabularies developed by agents other than DCMI.

Central to the idea of the DCAP is the idea that the DCAP does not itself declare new "terms", but rather references or "uses" (or reuses) "terms" that are declared elsewhere. The widespread adoption of XML has led to suggestions that the components used in XML languages are terms that can be referenced in DCAPs.

However, as discussed in detail in the previous section the names and components used in an XML language can not be deployed in an RDF graph or DC metadata description (except as XML Literals or rich representations), and since the very purpose of a DCAP is to specify the URI references that can occur in a DC metadata description, it follows that it is not meaningful to reference them in a DCAP. A DCAP can not "reuse XML elements".

If it is required for a DCAP to describe how to express some information that is currently expressed in an XML language, then it is necessary to develop a means of representing that same information within the framework of the DCMI Abstract Model and the RDF data model i.e. to establish a means of expressing the information that is currently represented using components within an XML tree structure in terms of the statement-based models of the DCAM and RDF.

That process is outlined in section 4 above. It involves either establishing how that information can be represented using an existing RDF vocabulary or developing a **new** RDF vocabulary i.e. identifying the set of properties, classes, and other resources required to express that information in the statement-based model, and providing URI references so that they can be referenced by DC metadata descriptions. Those new properties, classes and URI references are different things from the names and components used in the XML documents, and there may be no simple one-to-one correspondence between the names of components in the XML language and the URI references of the RDF vocabulary.

In order for those new terms to be useful to consumers of the data, and to be reused by the authors of other data, it is useful to provide descriptions of what those newly-coined URI references denote, either in the form of human-readable documentation, or machine-processable descriptions made using the RDF Schema language, or both.

If the entire XML language or some subset of the constructs used within an XML language is mapped into the DC/RDF data models, then documentation on that

mapping and/or tools that apply the mapping to XML documents (e.g. XSLT transforms) are valuable to other implementers. See also GRDDL [GRDDL].

The naming and ownership of the URI references of the new RDF vocabulary is a social-political question rather than a technical one, and it is important to distinguish this issue from the semantic modelling/mapping issue. To RDF, URI references are simply opaque strings. As the discussion in section 4 highlights, the properties and classes are different things from the components of an XML language and it is unlikely that there will be a simple one-to-one correspondence between them i.e. there will probably not be a simple correspondence between expanded names/QNames in the XML language and URI references in the RDF vocabulary. It may also be the case that the mapping and the RDF vocabulary is developed quite separately from the XML language, even by an agency that is not the owner of the XML language.

Whatever names are used within the XML language and the RDF vocabulary, it will be necessary to describe a *mapping between* the XML language and the RDF model and to be absolutely clear that the names (URI references) in the RDF vocabulary are different from the names (expanded names/QNames) in the XML vocabulary (see section 4.1).

It *may* be possible to select the URI references of the new RDF vocabulary so that when they are represented in RDF/XML as expanded names encoded as XML QNames, those names correspond to the expanded names and QNames used in the initial XML vocabulary. (But see Note 4: if a URI reference has been assigned to an XML element type, then that same URI reference can not also be assigned to an RDF property. However, bearing in mind a mapping is always required, and the content models for the named components in RDF/XML will be *different* from the content models of any component that uses that same name in the XML language - e.g. in the latter a sub-tree structure may be available - it may be preferable to ensure that URI references are chosen so that their XML expanded name/QName representation does *not* duplicate any of the names used in the vocabulary of the XML language. The LOM XML binding and the LOM RDF binding take this latter approach. There is *no* overlap between the expanded names/QNames used in the LOM XML binding and those used when a LOM RDF graph is serialised in RDF/XML. It is clear to the user that they are quite different XML vocabularies used in the context of two different XML languages (LOM XML and RDF/XML). That has no impact on the capacity to describe the mapping between the LOM XML language and the RDF data model.

However, the human users and owners of the two vocabularies may feel it is appropriate that the names in the two vocabularies carry some indication of a common source, e.g. that the URI references used in the RDF vocabulary are in some way similar to the URI references used as XML Namespace Names in the XML language.

5.4 Summary

- a DCAP specifies the "terms" that are referenced within a particular class of DC metadata description sets, and these "terms" are properties and classes
- a DCAP does not itself declare new "terms", but rather references or "uses" (or reuses) "terms" that are declared elsewhere
- the names and components defined by an XML language can not be deployed in a DC metadata description and so can not be referenced in a DCAP
- to express in a DC metadata description some information that is currently expressed in an XML language, it is necessary to carry out an analysis of the semantics of that XML language and to develop a means of representing that same information within the framework of the DCMI Abstract Model and the RDF data model, and to specify an RDF vocabulary to do that (either referencing an existing vocabulary or developing a new one)
- a new RDF vocabulary (if required) should be described in human-readable documentation and using the RDF Schema language
- the naming and ownership of any new RDF vocabulary is a separate issue from the semantic analysis/modelling/mapping issue. While it may be socially/politically advantageous that names in XML vocabulary and names in an RDF vocabulary carry some indication of common source/ownership, care must be taken to make it clear that these are different sets of names used in the context of different specifications, and there is a mapping between the XML language and the RDF/DC data model(s).

6. Conclusions / Recommendations

Ther is a good deal of confusion surrounding the concept of the DCAP and their construction. The absence of a clear specification of what a DCAP is, together with misunderstandings about XML and RDF, and some ambiguity in DCMI's use of terminology (or users' interpretation of that terminology), have meant that although the general idea of the DCAP has been widely embraced, in practice it has been interpreted and implemented in different ways, sometimes significantly different ways. Sometimes those interpretations and implementations are not consistent with the DCMI Abstract Model and/or with the data models used in the XML and RDF specifications.

This document has sought to highlight a small subset of the issues, particularly on the problem of of "reusing" or "mixing and matching" "terms". It has tried to clarify in some detail why an unqualified notion of "reuse" is problematic, with particular reference to XML and (in sections 5.3 and 5.4) to suggest how those problems might be addressed so that the conditions can be put in place to make the promise of "mixing and matching" realisable.

Notes

[1] This is a slight simplification, since XML documents can also contain other types of item such as comments and processing instructions, and these can form part of XML element content, but for the purposes of this document, we consider XML documents to be made up of XML elements and attributes.

[2] Again this is a simplification as not only are there other types of information item (see Note 1), but the InfoSet provides one "Character" information item for each character of element content: here a sequence of characters is represented as a single item. The Infoset also provides items related to the use of XML Namespaces.

[3] This is typically a human-readable document, though specifications like GRDDL [GRDDL] represent an attempt to disclose at least some of that information in machine processable form, by providing access to an XSLT transform (speciffic to that XML language) that generates an RDF/XML document from the XML document.

[4] A fragment of XML conforming to *any* XML language could, of course, be used as an RDF XML Literal (or a "rich representation" in the terms of the DCMI Abstract Model. In that case it is not interpreted by the RDF/XML parser; it is simply passed to the application as an XML fragment.

[5] The "CORES Resolution" [CORESRES] 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 frameworks and rely on those frameworks for their meaning and interpretation. The assignment of a URI reference to an "element" means that it can be unambiguously cited - and it could be the subject of a DC metadata description - but it does not change the nature of the "element": and it does not mean that it is meaningful to use that URI refererence as, e.g., a property URI in a DC metadata description. Indeed saying that a single URI reference denoted *both* an element defined within a hierarchical model *and* a property would contradict the principle that a URI should identify a single resource.

[6] I can't emphasise strongly enough how problematic it is that DCMI has no formal definition of what a DCAP actually is, and that documents like CEN CWA 14855 present a rather "loose" specification. I propose a notion of a DCAP here that seems consistent with most of the approaches to the idea that I've seen and which is based on the DC Abstract Model. But I readily admit I'm influenced by my own experience with various projects, and it is just one possible model for a DCAP!!! Someone else could propose a quite different, but equally valid, notion of a DCAP, also based on the Abstract Model (e.g. that a DCAP defined an application-specific XML language for representing DC metadata descriptions), and the arguments I make below would have to be modified for that case. (As an aside, I think a better name for what I describe here would be a DC Description Set Profile!)

References

[DCAPUB]
Thomas Baker, *DCMI Usage Board Review of Application Profiles*
http://dublincore.org/usage/documents/profiles/

[CWA14855]
CEN CWA14855 - Dublin Core Application Profile guidelines
http://www.cenorm.be/isss/cwa14855/

[DCMIAM]
DCMI Abstract Model
http://dublincore.org/documents/abstract-model/

[XML]
Extensible Markup Language (XML) 1.0 (Third Edition). W3C Recommendation 04 February 2004.
http://www.w3.org/TR/REC-xml

[XML]
XML Schema Part 0: Primer Second Edition. W3C Recommendation 28 October 2004.
http://www.w3.org/TR/xmlschema-0/

[XMLNS]
Namespaces in XML. W3C Recommendation 14 January 1999.
http://www.w3.org/TR/REC-xml-names

[XMLNS1.1]
Namespaces in XML 1.1. W3C Recommendation 04 February 2004.
http://www.w3.org/TR/xml-names11

[XMLINFO]
XML Information Set (Second Edition). W3C Recommendation 04 February 2004.
http://www.w3.org/TR/xml-infoset

[XPATH]
XML Path Language (XPath) Version 1.0. W3C Recommendation 16 November 1999.
http://www.w3.org/TR/xpath

[XQUERY]
XQuery 1.0: An XML Query Language. W3C Working Draft 11 February 2005.
http://www.w3.org/TR/xquery/

[XMLNS1.1]
[Editorial Draft] Versioning XML Languages. Proposed TAG Finding 16 November 2003.
http://www.w3.org/2001/tag/doc/versioning.html

[to be confirmed]

[DCXML]
Guidelines for implementing Dublin Core in XML
http://dublincore.org/documents/dc-xml-guidelines/

[RDFCAS]
RDF/XML Syntax Specification (Revised) W3C Recommendation 10 February 2004.
http://www.w3.org/TR/rdf-syntax-grammar/

[RDFCAS]
Resource Description Framework (RDF): Concepts and Abstract Syntax W3C Recommendation 10 February 2004.
http://www.w3.org/TR/rdf-concepts/

[RDFSEM]
RDF Semantics W3C Recommendation 10 February 2004.
http://www.w3.org/TR/rdf-mt/

[RDFS]
RDF Vocabulary Description Language 1.0 (RDF Schema) W3C Recommendation 10 February 2004
http://www.w3.org/TR/rdf-schema/

[DCMINS]
Namespace Policy for the Dublin Core Metadata Initiative (DCMI)
http://dublincore.org/documents/dcmi-namespace/

[GRDDL]
Gleaning Resource Descriptions from Dialects of Languages (GRDDL) W3C Coordination Group Note 13 April 2004
http://www.w3.org/TR/rdf-schema/

[CORESRES]
CORES Standards Interoperability Forum Resolution on Metadata Element Identifiers
http://www.cores-eu.net/interoperability/cores-resolution/

DCMI Mixing and Matching FAQ

Andy Powell
UKOLN, University of Bath

This document attempts to answer some of the practical questions that implementors ask when faced with a desire to incorporate their existing XML metadata semantics into DCMI metadata applications.

Question 1: My favorite XML schema contains an element type or attribute (e.g. my:price or my:currency) that I want to use in my Dublin Core metadata. How do I do it?

The bad news is that an existing XML element type or attribute **cannot** be used as is in DCMI metadata applications. This is a very important point, but is sometimes hard for people to understand. Before you can use your favorite element or attribute you must declare it as a DCMI-compatible term. The good news is that doing so need not be an overly onerous task. Here's what you have to do:

1. Decide whether your XML element type or attribute corresponds to DCMI's notion of a 'property' or an 'encoding scheme'. These notions are defined in the DCMI Abstract Model but, for convenience, the definitions are repeated here:

   A property is a specific aspect, characteristic, attribute, or relation used to describe resources.

   Encoding scheme is the generic name for vocabulary encoding scheme and syntax encoding scheme.
   A syntax encoding scheme indicates that the value string is formatted in accordance with a formal notation, such as "2000-01-01" as the standard expression of a date.
   A vocabulary encoding scheme is a class that indicates that the value of a property is taken from a controlled vocabulary (or concept-space), such as the Library of Congress Subject Headings.

2. Next, check whether an equivalent property or encoding scheme has already been defined by the DCMI (or elsewhere), for use in DCMI metadata. A good place to start checking is the list of DCMI Metadata Terms.
3. Next, assign a URI reference to your new property or encoding scheme - see question 3 below.
4. Finally, declare your new property or encoding scheme using the RDF Schema language (RDFS) and make this declaration available somewhere on the Web - see questions 4 and 5 below.

Question 2: Why can't I just re-use my XML element type or attribute as is?

Because XML element types and XML attributes are component parts of an XML language. Their meaning is determined solely by their placement in the XML tree structure of the given XML language and the semantics that the developers of that language chose to associate with that structure. On the other hand, DCMI properties and encoding schemes are conceptual entities within the DCMI Abstract Model - their meanings are defined by the model and by the semantic declarations that DCMI make available. Furthermore, XML element types and attributes are named using XML *expanded names* (a pair comprising an *XML Namespace Name* (which is a URI reference) and a *local name*). On the other hand, DCMI properties and encoding schemes are named using URI references.

So, although XML element types and DCMI properties may look superficially similar, for example `lom:title` looks similar to `dc:title` when the two are encoded in XML, in fact they are very different entities.

For those who are interested, the XML, RDF and DCAPs document provides a much more in-depth treatment of the differences

between XML element types and RDF properties and the usage of both in the context of DCMI metadata.

Question 3: How do I assign a URI to my 'element'?

Unfortunately, there is little best-practice in this area to draw on at the time of writing. The Guidelines for assigning identifiers to metadata terms document lists some possible approaches.

One immediate issue to consider is whether to make the URI reference that is assigned to the new property or encoding scheme similar to the XML expanded name of the XML element or attribute. DCMI has chosen to keep the two things very similar. For example, the XML expanded name that is used to represent the DC Title property according to the Guidelines for encoding DC in XML recommendation is `dc:title` (correspoinding to the `http://purl.org/dc/elements/1.1/` XML namespace name and the `title` local name). The DC Title property is assigned the `http://purl.org/dc/elements/1.1/title` URI reference. Therefore, the property URI reference is simply a concatenation of the component parts of the XML expanded name.

On the other hand, the RDF encoding of the IEEE LOM (which has effectively made the LOM available for use with DCMI metadata because the DCMI Abstract Model is essentially the same as the RDF model) has chosen to keep the XML expanded names used in the LOM/XML encoding and the URI references assigned to the LOM/RDF properties completely separate. *Example to be provided*

Remember that your new property is likely to appear in the various DCMI encoding syntaxes using the name that is the final part of the URI reference you assign (usually the bit after the final '/' or '#'). For example, the URI reference `http://example.com/my/terms#color` is likely to appear as `my:color` (in XML syntaxes) or `MY.color` (in HTML syntaxes).

If in doubt, register with the PURL system and assign a PURL to your new property or encoding scheme.

Question 4: How do I declare my XML element type or attribute as a DCMI property?

Use the RDF Schema language to do this. Take a look at the DCMI RDFS terms declarations for inspiration! As a minimum, you'll need something like this:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE rdf:RDF>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:dcterms="http://purl.org/dc/terms/"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">

  <rdf:Property rdf:about="http://purl.org/my/terms/price">
    <rdfs:label xml:lang="en-US">Price</rdfs:label>
    <rdfs:comment xml:lang="en-US">The amount of money needed to purchase the resource.
</rdfs:comment>
    <rdfs:isDefinedBy rdf:resource="http://purl.org/my/terms/"/>
    <dcterms:issued>2004-12-03</dcterms:issued>
    <dcterms:modified>2005-02-21</dcterms:modified>
    <dc:type rdf:resource="http://dublincore.org/usage/documents/principles/#element"/>
  </rdf:Property>
</rdf:RDF>
```

Make sure that this RDF/XML document is available (in this case) at both `http://purl.org/dc/terms/` and `http://purl.org/my/terms/price`.

Question 5: How do I declare my XML element type or attribute as a DCMI encoding scheme?

Use the RDF Schema language to do this. Take a look at the DCMI RDFS terms declarations for inspiration! As a minimum, you'll need something like this:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE rdf:RDF>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:dcterms="http://purl.org/dc/terms/"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">

  <rdfs:Class rdf:about="http://purl.org/my/terms/PoundsSterling">
    <rdfs:label xml:lang="en-US">Pounds Sterling</rdfs:label>
    <rdfs:comment xml:lang="en-US">A price in UK pounds sterling, formatted in the
following way: "P.pp"
    (where "P" representents one or more digits for the number of pounds and "pp"
represents
    two digits for the number of pence).</rdfs:comment>
    <rdfs:isDefinedBy rdf:resource="http://purl.org/dc/terms/"/>
    <dcterms:issued>2003-07-11</dcterms:issued>
    <dcterms:modified>2004-06-15</dcterms:modified>
    <dc:type rdf:resource="http://dublincore.org/usage/documents/principles/#encoding-
scheme"/>
  </rdfs:Class>
</rdf:RDF>
```

The examples used in questions 4 and 5 would allow the following XML fragment to be used in a DC/XML document that conforms to the Guidelines for implementing Dublin Core in XML recommendation:

```
<my:price xsi:type="my:PoundsSterling">2.99</my:price>
```

assuming that the appropriate namespace declarations are in place. Note that, by convention, the XML *local name* for an encoding scheme starts with an uppercase letter.

Question 6: I still don't understand! Do you have another example?

OK. Let's say that I want to start using DC metadata to describe car parts, and that my company (ZZ Motors) already uses an XML schema that allows for XML fragments like this:

```
<zz:carpart>
  <zz:engine>
    <zz:type>petrol</zz:type>
    <zz:capacity>2000cc</zz:capacity>
  </zz:engine>
  <zz:fueltank>
    <zz:capacity>25l</zz:capacity>
  </zz:fueltank>
<zz:carpart>
```

'zz' being associated with the `http://zz.com/carparts/` XML namespace name.

For the sake of argument, let's say that I want to start using a property in my DC metadata that indicates the engine capacity. Looking at my existing XML schema, I note that I already have an XML element type with the local name `capacity` (under `zz:engine`) that I might be able to use? But there's a problem! I also have an XML element type with the local name `capacity` elsewhere in my XML tree structure (under `zz:fueltank`). So I cannot simply use 'capacity' as the local name when I'm thinking about assigning a URI reference to my new property.

The semantics of my current XML element types called `capacity` are determined by the placement of those two element types in the XML tree. In fact I have two 'properties', which we'll call `engineCapacity` and `fueltankCapacity`. I'm interested in the first of these.

OK, so now I need to assign a URI reference to my new property called `engineCapacity`. I want this property to be widely used (because it'll make my supply chains work more smoothly if everyone else uses the same property) so I decide to name my

new property using a PURL, rather than a URI reference somewhere under my company's DNS domain name. I choose:

```
http://purl.org/carparts/terms/engineCapacity
```

Now I need to declare my new property using RDFS. I create a file on my company's Web site that contains the following:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE rdf:RDF>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:dcterms="http://purl.org/dc/terms/"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">

  <rdf:Property rdf:about="http://purl.org/carparts/terms/engineCapacity">
    <rdfs:label xml:lang="en-US">Engine Capacity</rdfs:label>
    <rdfs:comment xml:lang="en-US">The total combustion chamber size of an engine in
cubic centimetres.</rdfs:comment>
    <rdfs:isDefinedBy rdf:resource="http://purl.org/my/terms/"/>
    <dcterms:issued>2005-02-21</dcterms:issued>
    <dcterms:modified>2005-02-21</dcterms:modified>
    <dc:type rdf:resource="http://dublincore.org/usage/documents/principles/#element"/>
  </rdf:Property>
</rdf:RDF>
```

Finally, I register two PURLs, `http://purl.org/carparts/terms/` and `http://purl.org/carparts/terms/engineCapacity`, and configure them both so that they resolve to the RDF/XML document (above) on my Web server.

Content by: Andy Powell
Last updated: 22-Feb-2005

Date: 2005-02-09 15:43:14 - 01
From: Ann Apps <ann.apps@MANCHESTER.AC.UK>
Subject: XML schema
To: DC-LIBRARIES@JISCMAIL.AC.UK

At the DC-Library meeting in Shanghai I agreed to produce an XML
schema for the DCMI Library Application Profile (DCLibAP).

There is a first draft at:

http://epub.mimas.ac.uk/DC/dc-lib/xsd/dclib.xsd
...

Issues using MODS terms:

There are 3 MODS terms in the DCLibAP: location, edition and
dateCaptured.

edition and dateCaptured are sub-elements of mods:originInfo.
Thus I've included mods:originInfo in the XML schema and
commented out edition and dateCaptured. This means that
potentially you could use any other of the sub-elements from
originInfo as well even though they are not in the AP.

The way of encoding these MODS terms is different from, and not
really consistent with, DC practice to date. They have to include
nested elements, eg:

```
<mods:location>
  <mods:url>http://example.org/myurl</mods:url>
<mods:location>

<mods:originInfo>
  <mods:edition>Version 1</mods:edition>
</mods:originInfo>

<mods:originInfo>
  <mods:dateCaptured>2005-02-09<mods:dateCaptured>
<mods:originInfo>
```

Also mods:dateCaptured cannnot have an attribute
'xsi:type="dcterms:W3CDTF"'.

Date: 2005-02-09 23:59:49 - 02
From: Andy Powell <a.powell@ukoln.ac.uk>
Reply-To: DC-Libraries Working Group <DC-LIBRARIES@jiscmail.ac.uk>
To: DC-LIBRARIES@jiscmail.ac.uk
Subject: Re: XML schema

On Wed, 9 Feb 2005, Ray Denenberg, Library of Congress wrote:
> From: "Ann Apps" <ann.apps@MANCHESTER.AC.UK>
>> The way of encoding these MODS terms is different from, and not
>> really consistent with, DC practice to date. They have to include
>> nested elements, eg:
>>
>> <mods:location>
>>   <mods:url>http://example.org/myurl</mods:url>
>> <mods:location>
>
> Hi Ann,
>
> I'm not really up-to-date with dc so I don't know what dc practice you're
> referring to,

The fundamental problem here is that the XML encoding of DC is a representation of the underlying DC 'model' (the DCMI Abstract Model) which essentially is the same 'resource, property, value' triple model found in RDF. MODS (in common with many other XML-based languages, e.g. LOM, METS, etc.) does not share that same underlying model.

This is not a critisism of any of these other XML-based languages BTW - just a statement of fact.

It is therefore not possible to simply squish together DC 'elements' and MODS 'elements' in any kind of meangingful way.

Even if, as you suggest below, there is a way to make mods:url look superficially like a DC 'element' (i.e. to use it without the nesting), unless it really is a DC-compatible property, then what is being attempted here simply does *not* work.

By "DC-compatible property" I mean that the URI http://www.loc.gov/mods/v3url (*) identifies a 'property' (as defined in section 7 of the DCMI Abstract Model) that is intended to be used in the context of the DC Abstract Model and/or RDF. And, as a rule of thumb, I'd suggest this means that the semantics of this property should be declared using RDFS (as per all the current DC terms).

(*) Note the rather odd URI here caused by the mods namespace URI not ending in either a slash or a hash - I hope I've got this right. If not, apologies.

Andy.

> but it you want to be able to use
> <mods:url>http://example.org/myurl</mods:url>
> without wrapping it in <mods:location>, there's an easy way.
>
> The mads schema has the same problem, so we've created a "mods-for-mads"
> schema. It's totally compatible with the current mods (they produce
> identical instance sets) but a number of new data types have been created
> (url one of them) so that they can be directly referenced.
>
> Take a look at
> http://www.loc.gov/standards/mads/mads-preliminary-draft-2-dec-17.xsd
>
> It references mods url as:
> <xsd:element name="url" type="mods:urlType" .....
>
> And look at
> http://www.loc.gov/standards/mads/mods-for-mads.xsd
>
> It declares element <url> within <location> as
> <xsd:element name="url" type="urlType" minOccurs="0" maxOccurs="unbounded"
> />
> and creates a new definition urlType.
>
> So you could reference it as mods:url, You'd just need to change the
> schema location:
> <xsd:import namespace="http://www.loc.gov/mods/v3"
> schemaLocation="http://www.loc.gov/standards/mads/mods-for-mads.xsd" />
>
> Mads is still a draft. Our intention is to issue a new mods version (3.1)
> that includes these definitions, sometime after (or when) we release the
> first version of mads.
>
> We would be happy to include any other similar definitions in mods 3.1, if
> it makes sense to. (It was once suggested that we should treat every mods
> subelement in this fashion. I'm fairly sure we don't want to do that,
> because (1) it doesn't make sense in every case, and (2) it would create a
> much-less-readable schema.) For example, edition and dateCaptured of

> originInfo. If you'd like I'll change mods-for-mads so that these can be
> referenced (even though mads doesn't reference them currently) or any
> others.
>
> --Ray

Date: 2005-02-10 10:25:54 - 03
Reply-To: DCMI Architecture Group <DC-ARCHITECTURE@JISCMAIL.AC.UK>
Sender: DCMI Architecture Group <DC-ARCHITECTURE@JISCMAIL.AC.UK>
From: Andy Powell <a.powell@UKOLN.AC.UK>
Subject: Mixing and matching - not always! (was Re: XML schema (fwd))
To: DC-ARCHITECTURE@JISCMAIL.AC.UK

I'm forwarding a message from the dc-libraries list here, since it touches on an important architectural principle - namely, that you can not simply take an 'element' from an existing XML-based language like MODS or LOM and expect to be able to use it in a DC description. The fact that something exists as an XML Qname is not sufficient for it to be used as a property in DC.

Owners of such terms have to explicitly acknowledge that the terms are RDF properties (or at least declare them in such a way that they are able to be treated as RDF properties) before they can be used in DC application profiles. In practice, I suggest that this means that the semantics of these terms should be declared using RDFS.

Pete Johnston sent a follow-up to my message which explains this further...

--- cut ---
Yes. The real underlying problem is with the DC Libraries Application Profile.

It references DC "elements" and MODS "elements" as if they are the same type of thing, when in fact they are fundamentally different because (as Andy says) they are defined in the context of two different data models: DC "elements" are properties, and are defined in terms of the statement/triple-based DC Abstract Model and RDF data model; MODS "elements" are components in a hierarchical data structure, and their interpretation is defined in terms of that hierarchical data structure.

(And so it follows for example, that concepts like element containment ("sub-element", "child element"), which makes perfect sense in the hierarchical model, have no meaning in the DCAM/RDF models; and conversely notions like element refinement (subproperty) which is well defined in the DC/RDF models, have no place in the hierarchical model).

Any DC Application Profile has to be based on a single underlying data model, i.e. on the DC Abstract Model. The "mixing and matching" has to take place within the context of that framework, and this sort of "cross-model" hybridisation can not work.
--- cut ---

As an example of how this can work I would cite the MARC relator terms - where the Library of Congress have taken (are taking?) the time to explicitly re-declare an existing set of terms as RDF properties. Because this has been done, it is now (or very soon will be) possible to use the MARC relator terms in a DC application profile and for that usage to be maningful in terms of the DCMI Abstract Model.

Date: 2005-02-10 12:40:40 - 05
From: Rachel Heery <r.heery@UKOLN.AC.UK>
Subject: Re: Mixing and matching - not always! (was Re: XML schema (fwd)
Comments: cc: dc-libraries@jiscmail.ac.uk
To: DC-ARCHITECTURE@JISCMAIL.AC.UK

(sorry for X-posting, can WG chairs indicate on which list this discussion is best placed?)

On Thu, 10 Feb 2005, Andy Powell wrote:
> Owners of such terms have to explicitly acknowledge that the terms are RDF
> properties (or at least declare them in such a way that they are able to
> be treated as RDF properties) before they can be used in DC application
> profiles. In practice, I suggest that this means that the semantics of
> these terms should be declared using RDFS.

I think your bracketed statement needs more explanation... it would be helpful to be clear as to how terms can be 'declared in such a way' that they can be used as RDF properties. Even allowing for the constraints of the DC data model, there seems to me some wriggle room to enable mixing and matching where 'owners' of terms are willing to co-operate.

As I understand it the process for re-use of MARC relator terms was an initial agreement that (some of) the relator terms would be useful within DC records, then going through the formality of 'declaring' such terms as RDF properties - not trying to match the MARC data model to DC data model.
.....

> As an example of how this can work I would cite the MARC relator terms -
> where the Library of Congress have taken (are taking?) the time to
> explicitly re-declare an existing set of terms as RDF properties.
> Because this has been done, it is now (or very soon will be) possible to
> use the MARC relator terms in a DC application profile and for that usage
> to be maningful in terms of the DCMI Abstract Model.

I think it is the fact that the owner is willing to declare these terms 'outside' the rest of the MARC data model, as RDF properties that makes it ok to mix and match? within the MARC data model and MARC records the relator terms do not act as 'properties' as I understand it - the terms have a different role in MARC records than within DC records.

This seems to make declaring terms as RDF properties something of a formality - as long as the maintainer or 'owner' of data element sets is willing to declare a particular sub-set of terms as RDF properties then that is ok...

In my view the criteria for re-use of terms should be something like:

"First, are the semantics and context of a term in one metadata format sufficiently similar to the semantics and context of the property I want to express in a DC description? if so can this term be usefully used in 'isolation' within a DC description out of the context of its original format?

Second, are the 'owners' of the terms willing to co-operate?"

If the answer to both of the above is yes, then declaring those terms as RDF properties may well be achievable. Especially if, as I understand has happened with MARC relator terms, just the sub-set of terms required from the 'other' format based on a different data model need to be declared??

Maybe worth thinking about that old saying 'everything can be solved by a level of indirection'.... not knowing much about MODS, but could a sub-set of MODS terms be 'separated out' of MODS and declared as RDF properties?

In my view we should be looking for solutions to help us meet requirements of several user communities, and to move forward as regards the evolution of data element sets by allowing re-use of data elements. If this can be done by declaring sets of terms in RDFS then good....

Date: 2005-02-10 13:30:57 - 06

Reply-To: DCMI Architecture Group <DC-ARCHITECTURE@JISCMAIL.AC.UK>
Sender: DCMI Architecture Group <DC-ARCHITECTURE@JISCMAIL.AC.UK>
From: Andy Powell <a.powell@UKOLN.AC.UK>
Subject: Re: Mixing and matching - not always! (was Re: XML schema (fwd)

On Thu, 10 Feb 2005, Rachel Heery wrote:

> (sorry for X-posting, can WG chairs indicate on which list this discussion
> is best placed?)

Errr... I'm not sure to be honest.

> On Thu, 10 Feb 2005, Andy Powell wrote:
>
>>
>> Owners of such terms have to explicitly acknowledge that the terms are RDF
>> properties (or at least declare them in such a way that they are able to
>> be treated as RDF properties) before they can be used in DC application
>> profiles. In practice, I suggest that this means that the semantics of
>> these terms should be declared using RDFS.
>
> I think your bracketed statement needs more explanation... it would be
> helpful to be clear as to how terms can be 'declared in such a way' that
> they can be used as RDF properties. Even allowing for the constraints of
> the DC data model, there seems to me some wriggle room to enable mixing
> and matching where 'owners' of terms are willing to co-operate.
>
> As I understand it the process for re-use of MARC relator terms was an
> initial agreement that (some of) the relator terms would be useful within
> DC records, then going through the formality of 'declaring' such terms as
> RDF properties

That's correct - and is exactly what I am suggesting needs to happen in
every case where we want to re-use existing 'elements'.

> I think it is the fact that the owner is willing to declare these
> terms 'outside' the rest of the MARC data model, as RDF properties that
> makes it ok to mix and match?

Yes.

> within the MARC data model and MARC records
> the relator terms do not act as 'properties' as I understand it - the
> terms have a different role in MARC records than within DC records.

I think that perhaps 'different role' is open to an interpretation that is
too strong - but basically I agree with what you are saying here.
marc:artist has essentially the same semantics whether it is used in MARC
or in DC but it is being used in the context of different underlying
models.

> This seems to make declaring terms as RDF properties something of a
> formality - as long as the maintainer or 'owner' of data element sets is
> willing to declare a particular sub-set of terms as RDF properties then
> that is ok...
>
> In my view the criteria for re-use of terms should be something like:
>
> "First, are the semantics and context of a term in one metadata format
> sufficiently similar to the semantics and context of the property I want
> to express in a DC description? if so can this term be usefully used in
> 'isolation' within a DC description out of the context of its original
> format?
>
> Second, are the 'owners' of the terms willing to co-operate?"

Agreed on both counts - this is what I meant by 'explicitly acknowledge'

above.

> If the answer to both of the above is yes, then declaring those terms as
> RDF properties may well be achievable. Especially if, as I understand has
> happened with MARC relator terms, just the sub-set of terms required from
> the 'other' format based on a different data model need to be declared??

I think that all the MARC relator terms have been declared. But it
doesn't really matter - there would be no problem with only declaring a
sub-set.

> Maybe worth thinking about that old saying 'everything can be solved by a
> level of indirection'.... not knowing much about MODS, but could a sub-set
> of MODS terms be 'separated out' of MODS and declared as RDF properties?

Yes, that could happen. By 'separated out' I assume that you mean
assigned URIs that are different to the current MODS namespace URI?

One of the 'best-practice' issues that we need to think about is whether
the namespace URI associated with the mods:url used in MODS/XML should be
the same as the namespace URI associated with mods:url used in DC/XML (and
DC/RDF/XML)?

As an example, what I think Mikael has done with his RDF version of LOM is
to re-declare the LOM 'elements' as RDF properties using a different
namespace URI. These LOM/RDF properties become usable in DC descriptions
in a way that the original XML Qnames used in LOM/XML instances are not.

> In my view we should be looking for solutions to help us meet requirements
> of several user communities, and to move forward as regards the evolution
> of data element sets by allowing re-use of data elements. If this can be
> done by declaring sets of terms in RDFS then good....

Agreed.

Date: 2005-02-10 14:43:37 - 07
Reply-To: DCMI Architecture Group <DC-ARCHITECTURE@JISCMAIL.AC.UK>
Sender: DCMI Architecture Group <DC-ARCHITECTURE@JISCMAIL.AC.UK>
From: Pete Johnston <p.johnston@UKOLN.AC.UK>
Subject: Re: Mixing and matching - not always! (was Re: XML schema (fwd)

Quoting Rachel Heery <r.heery@ukoln.ac.uk>:

> within the MARC data model and MARC records
> the relator terms do not act as 'properties' as I understand it - the
> terms have a different role in MARC records than within DC records.

Yes.

> This seems to make declaring terms as RDF properties something of a
> formality - as long as the maintainer or 'owner' of data element sets is
> willing to declare a particular sub-set of terms as RDF properties then
> that is ok...

I think it is much more than a "formality", and personally I think it is
dangerous to think in terms of "(re)declaring" a (sub-)set of existing "terms"
as properties. If a "term" is a component in a hierarchical data structure then
that is what it is; that same "term" can not also be a property. e.g. an XML
element is not an RDF property (not even in RDF/XML).

I think this is what you are getting at in the first of your criteria below, but
I guess I just want to stress that it is problematic to go in search of
similarity where there are fundamental differences.

The work that has to be done is to consider how the _information_ represented
within the hierarchical data structure is to be represented within a

triple/statement-based model. There may be no simple one-to-one correspondence between the components of the hierarchical data structure and the components of the statement-based model.

Mikael Nilsson's paper(s) on the LOM RDF binding e.g.

http://rubens.cs.kuleuven.ac.be:8989/ariadne/CONF2003/papers/MIK2003.pdf

give an excellent account of this process for the case of the LOM. And emphasises that the translation must be done by looking at each component of the hierarchical model in turn

===
The container-based metamodel used by LOM is thus
not compatible with the metamodel used by Dublin
Core. When does this matter? Binding LOM to RDF is
the obvious example in this context, as the metamodel
of RDF is based on a property-value model and not containment.
In general, it leads to difficulties when trying
to combine terms from two metadata standards into the
same system. When the metamodels are compatible,
such a combination or mapping can be realized by simply
translating the metamodel contructs. If the metamodels
are incompatible, the translation must be done
on an idiosyncratic, element-by-element basis.
===

In Mikael's mapping, some LOM data elements are modelled as RDF properties - but the property and the LOM data element are still two different types of thing. In some cases two different LOM data elements are modelled using the same RDF property (describing two different resources). In other cases what are data element _values_ in LOM are modelled as RDF properties (e.g. the case of LOM Relation.Role); in other cases, there is quite substantial re-modeling required (e.g. the case of LOM Classification)

> In my view the criteria for re-use of terms should be something like:
>
> "First, are the semantics and context of a term in one metadata format
> sufficiently similar to the semantics and context of the property I want
> to express in a DC description? if so can this term be usefully used in
> 'isolation' within a DC description out of the context of its original
> format?
>
> Second, are the 'owners' of the terms willing to co-operate?"
>
> If the answer to both of the above is yes, then declaring those terms as
> RDF properties may well be achievable. Especially if, as I understand has
> happened with MARC relator terms, just the sub-set of terms required from
> the 'other' format based on a different data model need to be declared??
>
> Maybe worth thinking about that old saying 'everything can be solved by a
> level of indirection'.... not knowing much about MODS, but could a sub-set
> of MODS terms be 'separated out' of MODS and declared as RDF properties?

If MODS terms are components in a hierarchical data model, then those terms can not also be properties, IMHO. What has to happen is the sort of mapping between the models which Mikael describes for the LOM, and that can only be done by looking at the information represented by MODS data structures.

In effect this is the process that has taken place for the MARC relator codes, but it was a fairly trivial case, as by definition they represent types of relationship (between a resource and an agent) and fit neatly into the binary relation model of RDF. It's still taken an awfully long time though!

> In my view we should be looking for solutions to help us meet requirements
> of several user communities, and to move forward as regards the evolution
> of data element sets by allowing re-use of data elements. If this can be
> done by declaring sets of terms in RDFS then good....

But reuse has to happen within a consistent, coherent framework. The analogy I think I used at one point was Meccano parts and Lego bricks: I can build nice things with Meccano and I can build nice things with Lego.

But no matter how desperately I might want to reuse my nice funky bit of my Meccano spaceship in my Lego submarine, it wasn't designed to fit. If we try to encourage reuse regardless we'll end up with our submarines leaking and the nose cones falling off our spaceships.

Having said all this, and at the risk of sowing vile heresy....

... increasingly I do have more fundamental misgivings about the way we in DC have tended to approach this notion of "reuse".

In the RDF/DC triple/statement based model, properties and classes are defined as more or less independent stand-alone entities. Yes, we assert relationships between resources (subproperty, subclass etc) but I can use a URIref like http://purl.org/dc/elements/1.1/title to denote the concept of "having a title" quite independently from that of having a subject, identifier etc etc etc.

However, in XML-based applications like MODS, the component parts of the data structure do not have the same sort of independence/free-standing nature. MODS is an XML language or format, and the way individual components (XML elements, XML attributes) within MODS are interpreted is conditioned by their structural relationships with other components (containment relations, element/attribute relations etc) as defined by the rules of that XML language.

Now yes, if MODS had been developed as an RDF application, using a triple-based model, or if a full MODS RDF mapping was developed in the way that the LOM RDF mapping was developed, then the classes and properties would be available for use in DC metadata descriptions, and we could establish useful relations between DC properties and MODS properties and so on.

But the approach of "cherry-picking" particular parts of MODS and mapping only those particular bits to the RDF model, just because those particular bits of MODS _appear_ to be similar to something we might want to express in a DC description, and because we have the notion that reuse is an absolute, seems... well... it all starts to seem a bit bizarre, really!

What are we really achieving by doing this?

In the absence of a MODS RDF binding, what is anyone gaining by asking LoC to define two or three RDF properties called

http://www.loc.gov/mods/location

(and the other two or three things needed for the DC Lib AP - I've just guessed the URIrefs) picked pretty much from random parts of the MODS data structure.

It provides _no_ interoperability whatsoever between DC and MODS XML because we've just picked out some tiny part of the MODS data structure.

Why are we _insisting_ on "reuse" in this rather odd piecemeal sort of way, instead of simply declaring the properties required within DCMI vocabularies?

Date: 2005-02-10 15:03:17 - 08
Reply-To: DCMI Architecture Group <DC-ARCHITECTURE@JISCMAIL.AC.UK>
Sender: DCMI Architecture Group <DC-ARCHITECTURE@JISCMAIL.AC.UK>
From: Pete Johnston <p.johnston@UKOLN.AC.UK>
Subject: Re: Mixing and matching - not always! (was Re: XML schema (fwd)
Comments: To: Mikael Nilsson <mini@nada.kth.se>

Hi Mikael,

(Sorry, my last post crossed with yours and Andy's)

Quoting Mikael Nilsson <mini@nada.kth.se>:

> On Thu, 2005-02-10 at 13:30 +0000, Andy Powell wrote:
>
> > As an example, what I think Mikael has done with his RDF version of LOM is
> > to re-declare the LOM 'elements' as RDF properties using a different
> > namespace URI. These LOM/RDF properties become usable in DC descriptions
> > in a way that the original XML Qnames used in LOM/XML instances are not.
>
> Yes, this is what I did. In the original version I even mentioned that
> the binding was "dc-compatible", i.e. compatible with the then
> non-existent DCAM :-)

I guess I still think that process is rather more than "re-declaring" though - there is actually quite a lot of "re-modelling" involved in the LOM RDF mapping, looking at what information the LOM tree represents in terms of relations between resources, rather than the tree structure itself (e.g. the whole MetaMetadata thing, Relation.Role, Classification etc).

There is no necessary one-to-one mapping between an XML element in an XML tree-structure and an RDF property. You have to look beyond the tree-structure at the information which is being represented by that structure - unless you just want to create an RDF representation of the XML Infoset, (element-1 is-child-of element-2 and so on) which might be a satisfying academic exercise but doesn't get us very far ;-)

> Note that to use the URIs defined in the RDF version of LOM in an XML
> DCAP would be strange, to say the least, as it would be in conflict with
> the LOM XML binding. Unfortunately there is currently no solution to
> this conflict.

Yes. That's what I meant when I was saying XML elements and RDF properties are different things.

> I think the lesson here is that the DCAM is pretty useful, or indeed
> absolutely essential, and that the corresponding AMs of METS and LOM
> (the hierarchical models) are actually not as useful.
>
> An external entity that defines its terms so that they comply with the
> DCAM *OR* RDFS are actually on the safe side, METS and LOM do
> neither :-(

Right. Nor does MODS. :-(

Date: 2005-02-10 15:40:59 - 09
Reply-To: DCMI Architecture Group <DC-ARCHITECTURE@JISCMAIL.AC.UK>
Sender: DCMI Architecture Group <DC-ARCHITECTURE@JISCMAIL.AC.UK>
From: Mikael Nilsson <mini@NADA.KTH.SE>
Subject: Re: Mixing and matching - not always! (was Re: XML schema (fwd)

On Thu, 2005-02-10 at 13:30 +0000, Andy Powell wrote:

> As an example, what I think Mikael has done with his RDF version of LOM is
> to re-declare the LOM 'elements' as RDF properties using a different
> namespace URI. These LOM/RDF properties become usable in DC descriptions
> in a way that the original XML Qnames used in LOM/XML instances are not.

Yes, this is what I did. In the original version I even mentioned that
the binding was "dc-compatible", i.e. compatible with the then
non-existent DCAM :-)

Note that to use the URIs defined in the RDF version of LOM in an XML
DCAP would be strange, to say the least, as it would be in conflict with
the LOM XML binding. Unfortunately there is currently no solution to
this conflict.

I think the lesson here is that the DCAM is pretty useful, or indeed
absolutely essential, and that the corresponding AMs of METS and LOM
(the hierarchical models) are actually not as useful.

An external entity that defines its terms so that they comply with the
DCAM *OR* RDFS are actually on the safe side, METS and LOM do
neither :-(

Date: 2005-02-10 16:07:54 - 10
Reply-To: DCMI Architecture Group <DC-ARCHITECTURE@JISCMAIL.AC.UK>
Sender: DCMI Architecture Group <DC-ARCHITECTURE@JISCMAIL.AC.UK>
From: Rachel Heery <r.heery@UKOLN.AC.UK>
Subject: Re: Mixing and matching - not always! (was Re: XML schema (fwd)

On Thu, 10 Feb 2005, Pete Johnston wrote:
> In effect this is the process that has taken place for the MARC relator codes,
> but it was a fairly trivial case, as by definition they represent types of
> relationship (between a resource and an agent) and fit neatly into the binary
> relation model of RDF. It's still taken an awfully long time though!

Looking at MARC relator codes they ca be used in various ways in MARC
records, not necessarily in relation to an 'agent', they can be used with
'subjects' of a resource too e.g. with

600 $4 (Subject Added Entry -- Personal Name / Relator code)
610 $4 (Subject Added Entry -- Corporate Name / Relator code)
611 $4 (Subject Added Entry -- Meeting Name / Relator code)

see http://www.loc.gov/marc/relators/relators.html

I just mention this as it seems a point of difference in the way these
'properties' are use in DC as opposed to MARC. And I would say by re-using
MARC relator codes DC is 'cherry-picking' from MARC, which you denigrate
wrt re-use of MODS?

> But reuse has to happen within a consistent, coherent framework. The analogy I
> think I used at one point was Meccano parts and Lego bricks: I can build nice
> things with Meccano and I can build nice things with Lego.
>
> But no matter how desperately I might want to reuse my nice funky bit of my
> Meccano spaceship in my Lego submarine, it wasn't designed to fit. If we try to
> encourage reuse regardless we'll end up with our submarines leaking and the nose
> cones falling off our spaceships.
>

Nice analogy, but I don't think anyone is saying we encourage re-use
'regardless' of differences in formats, informed people are saying we
think these particular terms are equivalent in the way they are used, can
we do something about it??.

And taking your analogy a little further away from the well ordered
playroom where kids put their Meccano in one box and their Lego in
another... In digital library world metadata created using different
standards/models is exchanged between applications, and to do this is
converted more or less effectively. So just like little kids out there
bashing their toys together, throwing them into the wrong box and often
breaking them, conversions can be more or less 'lossy'. Toys are being
broken now, data is already getting lost on conversion.

The benefit of re-use is that the metadata creator, the owners of the
metadata formats and the world in general buy into an agreement 'we agree
these 2 data elements as more or less equivalent, we think you should do
the same'. This is as opposed to creating more and more conversion
programmes mapping between different data elements.

I would say piecemeal re-use is a step towards interoperability...

Date: 2005-02-10 16:20:43 - 11
Reply-To: DCMI Architecture Group <DC-ARCHITECTURE@JISCMAIL.AC.UK>
Sender: DCMI Architecture Group <DC-ARCHITECTURE@JISCMAIL.AC.UK>
From: Mikael Nilsson <mini@NADA.KTH.SE>
Subject: Re: Mixing and matching - not always! (was Re: XML schema (fwd)

On Thu, 2005-02-10 at 14:43 +0000, Pete Johnston wrote:

> ... increasingly I do have more fundamental misgivings about the way we in DC
> have tended to approach this notion of "reuse".
>
> In the RDF/DC triple/statement based model, properties and classes are defined
> as more or less independent stand-alone entities. Yes, we assert relationships
> between resources (subproperty, subclass etc) but I can use a URIref like
> http://purl.org/dc/elements/1.1/title to denote the concept of "having a title"
> quite independently from that of having a subject, identifier etc etc etc.
>
> However, in XML-based applications like MODS, the component parts of the data
> structure do not have the same sort of independence/free-standing nature. MODS
> is an XML language or format, and the way individual components (XML elements,
> XML attributes) within MODS are interpreted is conditioned by their structural
> relationships with other components (containment relations, element/attribute
> relations etc) as defined by the rules of that XML language.

This is very true. When I have worked on formalizing the LOM RDF binding
I have used the trick of trying to bring the *whole* context into the
definition of each RDF property, to make sure I don't loose any of the
semantics.

For example, there is an element Language in LOM, used in three places:
In the General category, in the Metametadata category, and in the
Educational category. Now if I had done the mapping naively, this would
be just one URI. But in reality it is two:

* dc:language is used for the General and Metametadata occurences, as
the semantics matches dc:language precisely, even though it describes
the language for two different resources (the learning object and its
metadata, respectively)
* lom_edu:language is used in the Educational category, as the element
means a slightly different thing (the intended primary language of the
user).

This is a simple example, but in general when mapping from hierarchical
models to RDF, one must be certain that all semantics hidden in the
context (in this case, the categories above the element) is brought into
the property definition.

In theory, this could lead to properties of the form:

lom_annotation:entity_name

if that were any different than for example:

lom_lifecycle:contribute_entity_name

It so happens that the semantics are identical, but the properties are
applied to different resources (the learning object and the annotation,
respectively), so only one URI is needed...

It goes to show that the mapping must indeed be done on an
element-by-element basis, and with _thorough_ knowledge of the semantics
of _each_ element/category.

Date: 2005-02-10 16:47:12 - 13
Reply-To: DC-Libraries Working Group <DC-LIBRARIES@JISCMAIL.AC.UK>
Sender: DC-Libraries Working Group <DC-LIBRARIES@JISCMAIL.AC.UK>
From: "Rebecca S. Guenther" <rgue@LOC.GOV>
Subject: Re: Mixing and matching - not always! (was Re: XML schema (fwd)

Some comments below.

On Thu, 10 Feb 2005, Rachel Heery wrote:

> (sorry for X-posting, can WG chairs indicate on which list this discussion
> is best placed?)
>
> On Thu, 10 Feb 2005, Andy Powell wrote:
>
> > >
> > > Owners of such terms have to explicitly acknowledge that the terms are RDF
> > > properties (or at least declare them in such a way that they are able to
> > > be treated as RDF properties) before they can be used in DC application
> > > profiles. In practice, I suggest that this means that the semantics of
> > > these terms should be declared using RDFS.
>
> I think your bracketed statement needs more explanation... it would be
> helpful to be clear as to how terms can be 'declared in such a way' that
> they can be used as RDF properties. Even allowing for the constraints of
> the DC data model, there seems to me some wriggle room to enable mixing
> and matching where 'owners' of terms are willing to co-operate.

Pete and Andy had agreed (as part of Usage Board work) to put together a
paper explaining better what this means, why MODS elements cannot be used
as RDF properties, and what needs to be done to be able to reuse MODS
elements. After all, those that are referenced in the DC-LAP are exactly
the semantics that were needed for the given element. I still don't
understand this completely.

> As I understand it the process for re-use of MARC relator terms was an
> initial agreement that (some of) the relator terms would be useful within
> DC records, then going through the formality of 'declaring' such terms as
> RDF properties - not trying to match the MARC data model to DC data model.
>
> ......
> .....
>
> > As an example of how this can work I would cite the MARC relator terms -
> > where the Library of Congress have taken (are taking?) the time to
> > explicitly re-declare an existing set of terms as RDF properties.
> > Because this has been done, it is now (or very soon will be) possible to
> > use the MARC relator terms in a DC application profile and for that usage
> > to be maningful in terms of the DCMI Abstract Model.
> >

And this was possible because we spent some time fitting our descriptions
of relator terms/codes into a form acceptable to UB members-- just
figuring out what to call the various elements that describe these
terms/codes (e.g. rdfs:label, rdfs:comment, etc.). Now our RDF expression
of relators is generated on the fly from our official documentation by
using stylesheets. It's a fairly mechanical process. And we didn't change
the list that we've been using for 30 or so years.

> I think it is the fact that the owner is willing to declare these
> terms 'outside' the rest of the MARC data model, as RDF properties that
> makes it ok to mix and match? within the MARC data model and MARC records
> the relator terms do not act as 'properties' as I understand it - the
> terms have a different role in MARC records than within DC records.
>
> This seems to make declaring terms as RDF properties something of a
> formality - as long as the maintainer or 'owner' of data element sets is

> willing to declare a particular sub-set of terms as RDF properties then
> that is ok...
>
> In my view the criteria for re-use of terms should be something like:
>
> "First, are the semantics and context of a term in one metadata format
> sufficiently similar to the semantics and context of the property I want
> to express in a DC description? if so can this term be usefully used in
> 'isolation' within a DC description out of the context of its original
> format?
>
> Second, are the 'owners' of the terms willing to co-operate?"

I would think in the case of these MODS elements the answer to both of
these is yes.

> If the answer to both of the above is yes, then declaring those terms as
> RDF properties may well be achievable. Especially if, as I understand has
> happened with MARC relator terms, just the sub-set of terms required from
> the 'other' format based on a different data model need to be declared??
>
> Maybe worth thinking about that old saying 'everything can be solved by a
> level of indirection'.... not knowing much about MODS, but could a sub-set
> of MODS terms be 'separated out' of MODS and declared as RDF properties?

Some of the MODS elements have equivalent DC elements. I suppose any such
subset would be those that are needed by an application profile?

In the case of Relators, we have an RDF expression of the whole list (as I
said above, generated on the fly) and only a subset has the statement that
it refines dc:contributor. We would need some guidance on how to do
this. Or perhaps there are tools to convert an XML schema to an RDF
one?

> In my view we should be looking for solutions to help us meet requirements
> of several user communities, and to move forward as regards the evolution
> of data element sets by allowing re-use of data elements. If this can be
> done by declaring sets of terms in RDFS then good....

Right, and this was the basis I think of Rachel's famous paper about
mixing and matching elements in different metadata schemas. Why redefine
something that has the same semantics if there's a way of just cooperating
instead?

Date: 2005-02-10 17:20:11 - 14
Reply-To: DCMI Architecture Group <DC-ARCHITECTURE@JISCMAIL.AC.UK>
Sender: DCMI Architecture Group <DC-ARCHITECTURE@JISCMAIL.AC.UK>
From: Pete Johnston <p.johnston@UKOLN.AC.UK>

Quoting Rachel Heery <r.heery@ukoln.ac.uk>:

> I just mention this as it seems a point of difference in the way these
> 'properties' are use in DC as opposed to MARC. And I would say by re-using
> MARC relator codes DC is 'cherry-picking' from MARC, which you denigrate
> wrt re-use of MODS?

Hehe heh, you read my mind - yes, in my first draft of that message, I was going
to include that case too! ;-)

The only real reason (IMHO) that the MARC relator properties have LoC URIs is
because they are "owned"/managed/administered by LoC, not by DCMI.

Also I think there is a difference between selecting the MARC relators and
selecting two/three of the many components of MODS, because - if they are
considered only as relations between resources and agents (and that is the ony
facet of their use that has been modelled in RDF) - the MARC relators _do_ form

a "self-contained" set in a way that the components of the MODS hierarchy do not (because of their interdependence with other components).

But yes, you are correct - LoC/DCMI has chosen to model only that one facet of the way the MARC relator codes are used in MARC.

> Nice analogy, but I don't think anyone is saying we encourage re-use
> 'regardless' of differences in formats, informed people are saying we
> think these particular terms are equivalent in the way they are used, can
> we do something about it??.
>
> And taking your analogy a little further away from the well ordered
> playroom where kids put their Meccano in one box and their Lego in
> another... In digital library world metadata created using different
> standards/models is exchanged between applications, and to do this is
> converted more or less effectively. So just like little kids out there
> bashing their toys together, throwing them into the wrong box and often
> breaking them, conversions can be more or less 'lossy'. Toys are being
> broken now, data is already getting lost on conversion.
>
> The benefit of re-use is that the metadata creator, the owners of the
> metadata formats and the world in general buy into an agreement 'we agree
> these 2 data elements as more or less equivalent, we think you should do
> the same'. This is as opposed to creating more and more conversion
> programmes mapping between different data elements.
>
> I would say piecemeal re-use is a step towards interoperability...

As long as our standards adopt different meta-models, then there is no alternative to this conversion. There _is_ no option for reuse.

A Lego brick can never be (re)used in a Meccano construction, and vice versa. I have to design the equivalent of my Lego nose cone using Meccano, and it will require me to start using Meccano parts and nuts and bolts (which I wouldn't use in Lego).

Similarly, the component in the hierarchical model can never be (re)used directly in the triple model. Rather, I have to analyse the information that is represented by a structure based on model A, and then create new components that can represent that same information in a structure based on model B - and as Mikael's examples from the LOM show, with hierarchical models, that analysis has to consider the entire data structure, not just one part of it.

Just to be clear - I'm not in principle objecting to having a property called

http://www.loc.gov/mods/location

owned and managed by LoC, and referenced by the DC Libraries Application Profile. I really don't care what URIs things have or who coins them, as long as they are persistent and I know what they denote so that I know how I should deploy them.

I'm just highlighting that the fact that that single property has a LoC/MODS URIref does not signify that it has anything to do with a component used within MODS XML. It is _not_ a "reuse" of the mods:location XML element defined within MODS XML; it is a property, a completely new thing, quite separate from the existing XML element. And the fact that it has that name does not create any sort of "interoperability" between DC Lib AP and the MODS XML format.

That "interoperability" would come from the development of an RDF mapping/binding for MODS (which might use MODS properties, MARC properties, DC properties, LOM properties, FOAF properties, etc etc etc).

So given that no RDF binding for MODS exists, (IMHO) the only reason for choosing to create a new property called

http://www.loc.gov/mods/location

rather than choosing to create a new property called

http://purl.org/dc/terms/location

is that presumably it would be "owned"/managed by LoC rather than by DCMI - and I have to admit it seems slightly odd (to me!) that we are considering asking LoC to do this - to coin a handful of properties, representing only two or three (fairly arbitrary) facets of the MODS information model.

Date: 2005-02-10 23:01:51 - 15
Reply-To: DC-Libraries Working Group <DC-LIBRARIES@JISCMAIL.AC.UK>
Sender: DC-Libraries Working Group <DC-LIBRARIES@JISCMAIL.AC.UK>
From: Pete Johnston <p.johnston@UKOLN.AC.UK>
Subject: Re: Mixing and matching - not always! (was Re: XML schema (fwd)

Quoting "Rebecca S. Guenther" <rgue@loc.gov>:

> Pete and Andy had agreed (as part of Usage Board work) to put together a
> paper explaining better what this means, why MODS elements cannot be used
> as RDF properties, and what needs to be done to be able to reuse MODS
> elements. After all, those that are referenced in the DC-LAP are exactly
> the semantics that were needed for the given element. I still don't
> understand this completely.

Yes, and my apologies that I haven't done this. I said to Andy and Ann that I'd try to get something done over the weekend and circulate it next week.

[snip]

> Some of the MODS elements have equivalent DC elements. I suppose any such
> subset would be those that are needed by an application profile?
>
> In the case of Relators, we have an RDF expression of the whole list (as I
> said above, generated on the fly) and only a subset has the statement that
> it refines dc:contributor. We would need some guidance on how to do
> this. Or perhaps there are tools to convert an XML schema to an RDF
> one?

No, this can't be done, or at least not in any generally useful way.

An XML Schema describes the structural constraints on a class of XML documents - it describes the XML tree structure, the "content models" for XML elements and XML attributes, which XML elements can be contained within which other XML elements and so on.

An "RDF Schema" (there's a camp that argues we shouldn't even use that terminology because of the confusion it causes ;-)) describes classes and properties and relationships between them.

They aren't alternative representations of thesame information - they are completely different things

As I was trying to say in my message last night, XML works with a hierarchical, container-based model - so in MODS, elements have attributes and child/sub-elements - but RDF is based on triples, simple "statements" asserting relationships between resources.

As Andy said, both models are good and useful, but they _are_ different, and the "components" in an XML document are completely different things from the "components" in an RDF graph.

> > In my view we should be looking for solutions to help us meet requirements
> > of several user communities, and to move forward as regards the evolution
> > of data element sets by allowing re-use of data elements. If this can be
> > done by declaring sets of terms in RDFS then good....

>
> Right, and this was the basis I think of Rachel's famous paper about
> mixing and matching elements in different metadata schemas. Why redefine
> something that has the same semantics if there's a way of just cooperating
> instead?

Yes, "mixing and matching" is a Good Thing _if_ the things which are mixed and matched are appropriate for "mixing and matching" ;-)

But trying to mix and match things which are in fact very different (because they have been defined/created in the context of different models/frameworks) simply doesn't work. (Over on dc-architecture, I used the analogy of Lego bricks and Meccano parts - both good and useful in their own context, but if I try to use them together, it doesn't work - my Meccano parts won't click and my Lego bricks can't be bolted).

Unfortunately our rather loose use of terminology - particularly words like "element" - has (IMHO) tended to encourage us to see similarities between things which are in fact very different. (The work on the Abstract Model is one means of trying to clarify this - we can now use that as a point of reference.)

In many cases it is better - indeed, absolutely necessary! - to define _new_ components which are appropriate for the different context of use - as indeed has been done in the case of the RDF properties that represent the MARC relator codes.

Date: 2005-02-11 09:24:43 - 16
Reply-To: DC-Libraries Working Group <DC-LIBRARIES@JISCMAIL.AC.UK>
Sender: DC-Libraries Working Group <DC-LIBRARIES@JISCMAIL.AC.UK>
From: "Rebecca S. Guenther" <rgue@LOC.GOV>
Subject: Re: Mixing and matching - not always! (was Re: XML schema (fwd)

At the time that mods:location was added to DC-LAP, we were in MODS version 2.1 and there were no subelements under <location>. In version 3.0 we decided to make the distinction in MODS between an identifier and a URL (electronic location), so we redefined location with 2 subelements: <url> and <physicalLocation>. The latter is equivalent to the previous version's <location>, i.e. it is for a repository that holds the resource. The location is DC-LAP is intended for the repository, so we had intended to make <physicalLocation> a global element. Developers of DC-LAP wanted to specify the institution that held the resource. There was no plan to use it for a URL.

When the Usage Board first considered the DC-LAP, the decision was made to include MODS elements because of the general guidelines that DCMI was following about reuse of metadata elements that already existed. Initially this discussion about XML elements vs. RDF properties didn't come up. So DC-LAP has included the MODS elements for quite some time. When the Collection Description proposal for IsAvailableAt came up, this is when the Usage Board started discussing the issue of using MODS elements, since it was clear that the proposed isAvailableAt had essentially the same semantics as mods:location. And it has been discussed a few times since then.

On a historical note, edition/version was almost included in the initial DC element set in 1995, but was thrown out because it was considered by some not to be "cross-domain, resource discovery" and too library centric. Since those criteria seem to be no longer the requirements to define a DC element, who knows if edition/version may creep in.

On Fri, 11 Feb 2005, Ann Apps wrote:
> Apologies for not explaining what I meant by MODS terms not
> being consistent with DC practice. But I think others have
> explained it for me :)
>
> Another (more simplistic) point about using the MODS terms, in

> particular mods:location.
>
> Looking at the MODS XML schema, mods:location has 2 sub-
> elements, according to the hierarchical XML model:
> physicalLocation and url. They are both optional and can occur
> many times (though physicalLocations must precede urls). Thus if
> the sub-elements were promoted to 'first class' elements (ie could
> be used without the surrounding <mods:location> tags) (and
> assuming it were possible to sort out the definition in RDF terms)
> you would end up with 2 location properties rather than a single
> one.
>
> I understood that mods:location was included in the DC-Lib AP so
> that either a physical location or a digital location (or both) could be
> captured within the same property, not to have separate properties.
>
> The situation at the moment is that you need to write:
> <mods:location>
> <mods:physicalLocation>My Library</mods:physicalLocation>
> </mods:location>
> for a physical location and:
> <mods:location>
> <mods:url>http://example.com/mylibrary.</mods:url>
> </mosd:location>
> for a digital location
> [<mods:location>My Library</mods:location> is wrong according
> to the schema.]
>
> Whereas if there were a DCMI property for location, like all DC
> properties, it's value could be represented by either a URL or a text
> string.
>
> It also seems to me that a location property must bear some
> similarity to that needed but not yet decided by the Collection
> Description AP (isAvailableAt, isLocatedAt?).
>
> As for 'edition', this sounds akin to my long-lamented 'version'
> property :). But, joking aside, I do think there is a general need for
> a DCMI edition/version property (I seem to remember a question on
> askDCMI not long ago).

Date: 2005-02-11 13:50:54 - 17
From: Ann Apps <ann.apps@MANCHESTER.AC.UK>
Subject: Re: Mixing and matching - not always! (was Re: XML schema (fwd)
To: DC-LIBRARIES@JISCMAIL.AC.UK

Apologies for not explaining what I meant by MODS terms not
being consistent with DC practice. But I think others have
explained it for me :)

Another (more simplistic) point about using the MODS terms, in
particular mods:location.

Looking at the MODS XML schema, mods:location has 2 sub-
elements, according to the hierarchical XML model:
physicalLocation and url. They are both optional and can occur
many times (though physicalLocations must precede urls). Thus if
the sub-elements were promoted to 'first class' elements (ie could
be used without the surrounding <mods:location> tags) (and
assuming it were possible to sort out the definition in RDF terms)
you would end up with 2 location properties rather than a single
one.

I understood that mods:location was included in the DC-Lib AP so
that either a physical location or a digital location (or both) could be

captured within the same property, not to have separate properties.

The situation at the moment is that you need to write:
<mods:location>
  <mods:physicalLocation>My Library</mods:physicalLocation>
</mods:location>
for a physical location and:
<mods:location>
  <mods:url>http://example.com/mylibrary.</mods:url>
</mosd:location>
for a digital location
[<mods:location>My Library</mods:location> is wrong according
to the schema.]

Whereas if there were a DCMI property for location, like all DC
properties, it's value could be represented by either a URL or a text
string.

It also seems to me that a location property must bear some
similarity to that needed but not yet decided by the Collection
Description AP (isAvailableAt, isLocatedAt?).

As for 'edition', this sounds akin to my long-lamented 'version'
property :). But, joking aside, I do think there is a general need for
a DCMI edition/version property (I seem to remember a question on
askDCMI not long ago).

Date: 2005-02-15 09:33:46 - 18
Reply-To: DCMI Architecture Group <DC-ARCHITECTURE@JISCMAIL.AC.UK>
Sender: DCMI Architecture Group <DC-ARCHITECTURE@JISCMAIL.AC.UK>
From: Andy Powell <a.powell@UKOLN.AC.UK>

On Thu, 10 Feb 2005, Pete Johnston wrote:

> What are we really achieving by doing this?
>
> In the absence of a MODS RDF binding, what is anyone gaining by asking LoC to
> define two or three RDF properties called
>
> http://www.loc.gov/mods/location
>
> (and the other two or three things needed for the DC Lib AP - I've just guessed
> the URIrefs) picked pretty much from random parts of the MODS data structure.
>
> It provides _no_ interoperability whatsoever between DC and MODS XML because
> we've just picked out some tiny part of the MODS data structure.
>
> Why are we _insisting_ on "reuse" in this rather odd piecemeal sort of way,
> instead of simply declaring the properties required within DCMI vocabularies?

...and then later...

> So given that no RDF binding for MODS exists, (IMHO) the only reason for
> choosing to create a new property called
>
> http://www.loc.gov/mods/location
>
> rather than choosing to create a new property called
>
> http://purl.org/dc/terms/location
>
> is that presumably it would be "owned"/managed by LoC rather than by
> DCMI - and I have to admit it seems slightly odd (to me!) that we are
> considering asking LoC to do this - to coin a handful of properties,
> representing only two or three (fairly arbitrary) facets of the MODS
> information model.

At a technical level, I agree with you - it doesn't really matter whether a new term is assigned a DC URI or a MODS/LoC URI. And there are certainly some advantages (simplicity being the prime one) in favour of taking the terms we are interested in and plonking them into a DC namespace (i.e. assigning them a DC URI).

But, IMHO, the reasons for promoting the use of LoC, LOM and other URIs for existing terms has to do with the fuzzier social and political benefits this brings in terms of ownership, buy-in by the community and so on. I don't think we'll get buy-in to the DCMI approach by effectively saying to those communities, "We liked your XML element so much, we've taken a copy of it and added it to a DCMI namespace"!?

Instead, we need to find ways of explaining to them the benefits of the semantic Web approach. We've got to explain why we need an agreed underlying model (such as that provided by RDF) before we can mix and match metadata terms in the way we want. We've got to explain why an approach based on simply merging together lots of XML fragments doesn't scale beyond very limited cases where you've got explicit agreement of all the parties. We need to convince these other communities that it is worth their while re-casting the semantics of their metadata terms in an RDF context.

My guess is that other communities will want to feel ownership of the metadata terms that they create (just as we feel most comfortable with dc:title being dc:title and not being redeclared as somethingelse:title). And they'll only feel comfortable with DCMI if they think we are giving more than we are taking. In any case, I don't see that DCMI particularly wants to lumber itself with maintaining a whole set of metadata terms that are already defined and used by other communities.

All in all, I think we (primarily the Architecture WG and the Usage Board) need to explain not only the 'Whys' above, but also the 'Hows'. How do I declare my XML element as a property in RDF? How do I assign a URI to my term? How do I declare my controlled vocabulary as a DCMI encoding scheme? Etc, etc. I know that Pete is already working on some of the documentation that will help to do this.

None of which is easy on a shoestring! :-(

However, I do agree with you that the metadata 'application profile' notion of mixing and matching has tended to be interpretted much too simplisticly - and outside of its original context of the semantic Web - to mean "as long as it's in XML it must be OK" pretty much! But next time someone says to me that they've got an XML element that they're going to use as a DCMI property, rather than saying "you can't do that" (which is probably what it sounded like I said this time!) I'm going to try saying, "OK, but in order to do that, you need to do the following...".

Well, maybe... :-)

Element Refinement in Dublin Core Metadata

Creator:	Pete Johnston
Date Issued:	2005-04-11
Identifier:	http://dublincore.org/documents/2005/04/11/dc-elem-refine/
Replaces:	http://dublincore.org/documents/2005/01/23/dc-elem-refine/
Is Replaced By:	Not applicable
Latest Version:	http://dublincore.org/documents/dc-elem-refine/
Status of Document:	This is a draft DCMI Recommended Resource.
Description:	This document describes the concept of "element refinement" as used in Dublin Core metadata. It seeks to explain the consequences of stating that one property "refines" a second property. The purpose is to clarify that in some cases it may be appropriate and useful to make such an assertion and in other cases such an assertion may result in contradictions.

Elements, Properties and Statements

Dublin Core elements and element refinements are properties. A property is "a specific aspect, characteristic, attribute, or relation used to describe resources". According to the *DCMI Abstract Model*, all these "aspects, characteristics, attributes and relations" involve relationships between resources [DCMIAM]. Each property corresponds to a type of relationship, such as the notion that a resource "is created by" an agent (the agent is a second resource), or that a resource "is about the subject of" some concept (the concept is also a resource).

A property is itself a resource, a "conceptual" resource. When DCMI adds a property to one of its "DCMI Namespaces", it creates a human-readable description of that concept, and it assigns a globally unique identifier to the property, in the form of a URI.

The scope of URIs is global: the URI is used as if it denotes that same concept, that same relationship type, wherever it is cited. Further, the persistence policies described in the *Namespace Policy for the Dublin Core* guarantee that the URIs DCMI assigns to its properties will always denote that same essential concept [DCMINS]. So, the assignment of a URI means that other parties can use this unique identifier, and the combination of its global uniqueness and persistence mean that the reference is unambiguous.

The URI assigned to the property can be used in statements in Dublin Core metadata descriptions. According to the *DCMI Abstract Model*, a DC metadata description is a set of one or more statements about a single resource, and a statement is a two-part construct consisting of a reference to a property and a reference to a second resource, a value [DCMIAM]. The reference to the subject of the description is made by a URI (the "resource URI"). The reference to a property also takes the form of a URI (the "property URI"). The reference to the value may take the form of a URI or a "value representation". For the purposes of this discussion, the examples show the simplest cases where that reference takes the form of a URI (a "value URI") or a representation in the form of a string (a "value string").

Each statement asserts that a relationship of the type indicated by the property exists between two resources: the resource that is the subject of the description, and the value (see note [1]):

Resource URI	Statements

	Property URI	Value URI
ex:book1	dc:subject	ex:SemanticWeb

Element Refinement

In addition to providing a definition and identifier for each of the properties it declares, DCMI also describes relationships between these properties. If the definitions of two properties are such that whenever two resources are related by the first property they are also related by the second property, the first property is said to "refine", or to be a "refinement" of, the second property.

So for example, the definition of the property `dcterms:created` is "Date of creation of the resource", and the definition of the property `dc:date` is "A date associated with an event in the life cycle of the resource". The date of creation of a resource is always "a date associated with an event in the lifecycle of the resource", so the `dcterms:created` property refines the `dc:date` property.

The machine-processable schemas published by DCMI include descriptions of all DC elements and element refinements. In the description of an element refinement, a statement is included using the property URI `rdfs:subPropertyOf` from the RDF Vocabulary Description Language (RDF Schema) [RDFS]. This states that a relationship exists between two properties, and the nature of that relationship is defined by the RDFS concept `rdfs:subPropertyOf`:

> The property rdfs:subPropertyOf is an instance of rdf:Property that is used to state that all resources related by one property are also related by another [RDFS].

This `rdfs:subPropertyOf` assertion enables a human reader or a software application to infer new information when they encounter a statement made using the "refinement" or subproperty.

So if it is asserted that

```
dcterms:created rdfs:subPropertyOf dc:date
```

and a statement is made using `dcterms:created` as a property URI, e.g.

Resource URI	Statements	
	Property URI	Value String
ex:book1	dcterms:created	"1973-05-05"

then it can be inferred that it is also true that

Resource URI	Statements	
	Property URI	Value String
ex:book1	dc:date	"1973-05-05"

This outcome holds for **all** statements made using the URI of the element refinement as a property URI: whenever two resources are related by the element refinement, they are also related by the corresponding element. So an assertion that one element refines another - that an `rdfs:subPropertyOf` relationship exists between the properties - should be made only when the definitions of the properties support that conclusion.

"To refine or not to refine"

If there is just one case where the inferred statement would not be true, then the refinement/subPropertyOf relationship should not be asserted.

Consider, for example, the case of the properties `marcrel:OWN` ("The person or organization that currently owns an item or collection") and `dc:contributor` ("An entity responsible for making contributions to the content of the resource"). Both properties describe relationship types that relate a resource to an "entity", an agent capable of some action. (The `marcrel:OWN` property is part of a set of

properties defined by the Library of Congress, based on the MARC Relator Codes [MARCREL].)

And for a specific resource, it may well be true that a single entity is both an owner of and a contributor to that resource. But that does not apply in **all** cases. i.e. there are some resources where the entity who is the owner of the resource has not made a contribution to the content of the resource: not all resource owners are resource contributors. If `marcrel:OWN` was described as a refinement of `dc:contributor`, then that would mean that **every** statement using `marcrel:OWN` as a property URI would result in a statement using `dc:contributor` as a property URI, which would not be desirable.

Note also that the **absence** of a subproperty assertion in no way limits the capacity of the metadata author to say that, **for any given resource**, the same entity is both the owner and the contributor. The metadata author simply makes the two statements separately:

Resource URI	Statements	
	Property URI	**Value URI**
`ex:book1`	`dc:contributor`	`ex:agent1`
	`marcrel:OWN`	`ex:agent1`

As a second example, consider the case of `dc:date`, defined as "A date associated with an event in the life cycle of the resource". If an implementer uses a property `exterms:updatingFrequency` to indicate "The periodicity of modifications to the resource", and describes that property as an element refinement of `dc:date`, then statements such as the following might be inferred:

Resource URI	Statements	
	Property URI	**Value String**
`ex:document1`	`dc:date`	`"Monthly"`

Resources which are appropriate values for statements using `exterms:updatingFrequency` are not appropriate values for statements using `dc:date`, so it is **not** appropriate to describe that property as an element refinement of `dc:date`.

Similarly, consider `dc:rights`, defined as "Information about rights held in and over the resource". Suppose an implementer uses a property `exterms:privacyIndicator` to indicate whether a document should be publicly available or not, and specifies that Boolean (yes/no) values should be used. If that property is described as an element refinement of `dc:rights`, that would result in statements such as the following being inferred:

Resource URI	Statements	
	Property URI	**Value String**
`ex:document1`	`dc:rights`	`"Yes"`

There may be an argument that strictly speaking a Boolean value does not contradict the definition of `dc:rights`, but it would be difficult to consider the value "Yes" to be "information about rights held in and over the resource". So, again, it is **not** appropriate to describe `exterms:privacyIndicator` as an element refinement of `dc:rights`, because of the statements that would be inferred.

"How many?" : Multiple Refinement Relationships

In the declarations that DCMI makes, any given property is the subject of only one `rdfs:subPropertyOf` relationship: a DC element refinement refines exactly one element - though an element may be refined by multiple element refinements.

However, in principle, multiple assertions might be made, with the result that when the property is used in a statement **multiple** additional relationships can be inferred to exist.

So if it is asserted that

```
exterms:bookDistributor rdfs:subPropertyOf exterms:distributor
```

and also

```
exterms:bookDistributor rdfs:subPropertyOf dc:publisher
```

and a statement is made using `exterms:bookDistributor` as a property URI, e.g.

Resource URI	Statements	
	Property URI	Value URI
`ex:book1`	`exterms:bookDistributor`	`ex:Company1`

then the following **two** statements are also true:

Resource URI	Statements	
	Property URI	Value URI
`ex:book1`	`exterms:distributor`	`ex:Company1`
	`dc:publisher`	`ex:Company1`

Note that it is **not** a question of choosing one option over the other, or two applications behaving in different ways: **both** statements are implied in **all** cases, and as long as the two applications have "knowledge" of the two subproperty relations, they should both generate the same inferences.

"Who says so?": DCMI Namespaces and other Metadata Vocabularies

The capacity to assert the existence of `rdfs:subPropertyOf` relationships involving properties from the DCMI Namespaces is not limited to DCMI.

The publisher of another vocabulary may wish to declare that a property in that vocabulary is a subproperty of a property from the DCMI Namespaces (or even that a property from the DCMI Namespaces is a subproperty of a property from their vocabulary).

The Library of Congress defines a set of properties based on the MARC Relator Codes [MARCREL], that can be used to assert relationships between resources and agents. It is useful that, where appropriate, subproperty relations between these properties and properties from the DCMI Namespaces are declared. e.g.

```
marcrel:ARR rdfs:subPropertyOf dc:contributor
```

(`marcrel:ARR` denotes a property which links a musical composition and its arranger.)

With such information available, a Dublin Core application that encounters

Resource URI	Statements	
	Property URI	Value URI
`ex:music1`	`marcrel:ARR`	`ex:person1`

can derive the statement

Resource URI	Statements	
	Property URI	Value URI
`ex:music1`	`dc:contributor`	`ex:person1`

This means that an application that has no "prior knowledge" of `marcrel:ARR`, but which does derive the appropriate inferences from the assertion of the `rdfs:subPropertyOf` relationship, can make use of the `dc:contributor` statement. Such a process enhances interoperability between metadata applications.

A subproperty assertion may be made by a third party who is not the owner/publisher of either of the properties involved. Suppose a

metadata vocabulary has been constructed by a designer with no knowledge of the existence of Dublin Core. In their descriptions of their properties they have made no references to the notion that their property `exterms:songwriter` is related to `dc:creator`. If an implementer is working with both that metadata vocabulary and with the DCMI vocabularies, it may be useful for them to make a subproperty assertion:

```
exterms:songwriter rdfs:subPropertyOf dc:creator
```

so that, when their Dublin Core application encounters

Resource URI	Statements	
	Property URI	Value URI
`ex:music1`	`exterms:songwriter`	`ex:person1`

that application can derive the statement

Resource URI	Statements	
	Property URI	Value URI
`ex:music1`	`dc:creator`	`ex:person1`

Finally, it should be re-emphasised that the existence of a subproperty relationship with a DC property is not a requirement for the use of properties from other metadata vocabularies in DC metadata. In the case of the MARC Relator properties, a subproperty relation may exist in many cases, but those properties for which no such relation exists, such as the owner property `marcrel:OWN`, can still be deployed in statements in DC metadata descriptions.

Summary

- A declaration that one property refines, or is a subproperty of, a second property is an assertion of a **specific type of relationship** between the two properties.
- The meaning of the refinement/subproperty relationship is defined by the RDF Vocabulary Description Language (RDF Schema): the existence of a subproperty assertion states that **all resources related by one property are also related by the second property**.
- A subproperty assertion is **global**: it applies to **all** occurrences of the property URI in a statement; care should be taken not to assert subproperty relations where the inferences that may be drawn are incorrect or contradictory.
- Refinement/subproperty relations may be asserted to exist between **any** two properties.
- A property may be the subject of **multiple** refinement/subproperty relations.
- The existence of subproperty assertions about a property enables an application to infer **additional statements** from statements made using that property; the availability of subproperty assertions supports semantic interoperability between applications
- A subproperty assertion is **not a requirement** for using a property from another vocabulary in Dublin Core metadata descriptions.
- Subproperty assertions may be made by the owner(s) of the properties or by a third party.

Notes

[1] For the sake of brevity, in the examples, URIs are represented by Qualified Names. Prefixes are assumed to be associated with Namespace Names as follows:

- **dc:** `http://purl.org/dc/elements/1.1/`
- **dcterms:** `http://purl.org/dc/terms/`
- **dcmitype:** `http://purl.org/dc/dcmitype/`
- **marcrel:** `http://www.loc.gov/marc.relators/`
- **rdf:** `http://www.w3.org/1999/02/22-rdf-syntax-ns#`
- **rdfs:** `http://www.w3.org/2000/01/rdf-schema#`
- **exterms:** `http://example.org/terms/`
- **ex:** `http://example.org/things/`

N.B. At the time of writing, the URIs to be assigned to the MARC Relator properties are still under discussion. The URIs used here are provisional, and should not be cited without first checking the Library of Congress Web site for the authoritative identifiers of the properties.

References

[DCMIAM]
DCMI Abstract Model
http://www.ukoln.ac.uk/metadata/dcmi/abstract-model/

[DCMINS]
Namespace Policy for the Dublin Core Metadata Initiative (DCMI)
http://dublincore.org/documents/dcmi-namespace/

[MARCREL]
MARC Code List: Part I: Relator Codes
http://www.loc.gov/marc/relators/relators.html

[RDFS]
RDF Vocabulary Description Language 1.0 (RDF Schema)
http://www.w3.org/TR/rdf-schema/

Valid XHTML 1.0! Valid CSS!

Metadata associated with this resource: http://dublincore.org/documents/dc-elem-refine/index.shtml.rdf

Title: Comments on PB Core
Creator: DCMI Usage Board
Date: 2004-06-22

At its meeting in Bath, UK in March 2004, the DCMI Usage Board discussed the metadata element set "PB Core", which was developed for the public broadcasting community on the basis of Dublin Core metadata elements [1]. The Usage Board considers application profiles in terms of their conformance with Dublin Core principles [2]. The following is not a thorough review of PB Core but an overview of aspects which are problematic from the standpoint of an application profile conforming to DCMI principles.

There are several areas in which PB Core diverges from established DCMI principles and practices:

1. Use of the "dot" syntax to form metadata element names. This naming style was used in HTML implementations of Dublin Core of the late 1990s but has been superseded by a naming style (and data model) which considers both elements and element refinements as "properties" -- and accordingly with names that stand on their own (e.g., see Footnote 9 of [3], the "DCMI Policy on Naming Terms" [4], and the section on Element Refinements in [5]). In PB Core, then "Title.Program" would be "programTitle".

2. Element Refinements must refine -- but not extend -- the semantics of an element (see [2]). Some of the element refinements proposed in PB Core, however, change or broaden the meaning of the base element. Examples are:

   Title.Series

   > The semantics of dc:title are: "Definition: A name given to the resource. Comment: Typically, Title will be a name by which the resource is formally known." "The resource" means here: "the resource being described by dc:title".
   >
   > However, pbcore:Title.Series does not refer to "the resource being described by dc:title". Rather, it refers to the title of the series to which the resource being described by dc:title belongs.
   >
   > In this case, the Series may be seen as a separate entity -- it is in effect a "related resource" (see the draft Abstract Model [6] for a discussion of "related resource"). The relationship between a described resource and a related resource may be described with dc:relation. The DC Libraries Working Group discussed such a case in the context of the DC-Lib Application Profile and recommended pointing to related resources such as Series with the term dct:isPartOf, a refinement of dc:relation [7]. (In principle, one might coin an even narrower term such as pbs:isPartOfSeries, but such a local refinement would presumably not be understood outside of PBS.)

   Language.Usage

   > The semantics of dc:language are: "Definition: A language of the intellectual content of the resource. Comment: Recommended best practice is to use RFC 3066 [RFC3066] which, in conjunction with ISO639 [ISO639]), defines two- and three-letter primary language tags with optional subtags. Examples include "en" or

"eng" for English, "akk" for Akkadian", and "en-GB" for English used in the United Kingdom."

In the draft PB Core, the intended use of Language.Usage is to record the existence and type of other audio and textual representations of the main audio or language presentation mode for a resource or asset. In the draft PB Core, suggested values for the element include controlled terms such as "DVD Subtitle01".

However, while "eng" ("English") is a proper value of dc:language, "DVD Subtitle01" is not. Rather, the PB Core drafters mean to say, in effect: "This resource (a DVD) also has subtitles, which have the language English."

In modeling terms, one might either describe each set of subtitles or captions as a "related resource" with attributes of its own (such as dc:language); this solution would be formally precise but complex and expensive to implement. Alternatively, one might more simply see the languages of these multiple component parts all as attributes of the DVD as a whole. The metadata would then say, in effect, "This DVD has the languages English, Spanish, and Portuguese" without distinguishing between subtitles and captions.

These two options involve trade-offs between simplicity (which in this case would be simplistic) and complexity (which would be expensive), with implications for interoperability. One compromise solution would be to create PBS-specific refinements such as "Subtitle Language" or "Caption Language", each of which would hold "a language of the resource" (i.e., a language code) and thus dumb down properly to Language.

Applications outside of PBS -- unaware perhaps of these distinctions -- might interpret these different refinements in an undifferentiated way as meaning "This DVD has the languages English, Spanish, and Portuguese" (see above). In other words, the distinctions between subtitles and captions might be lost to non-PBS users of PBS metadata. This approach could work, however, if it is sufficient for PBS that the distinctions be understood internally.

3. Some PB Core "element refinements" are not element refinements in the sense of the DCMI model. Examples are:

   Relation.Type
   Creator.Role

   Both cases in effect imply a data model in which a Dublin Core element -- e.g., dc:relation or dc:creator -- itself has an attribute which, in turn, has a value in which controlled vocabularies (e.g., "IsPartOf" for Relation, "Cinematographer" for Creator) are used.

   In the DCMI model [1,5], however, terms such as IsPartOf or Cinematographer are themselves "element refinements" with respect to elements. In other words, the term identified by the URI http://purl.org/dc/terms/isPartOf semantically refines the term http://purl.org/dc/elements/1.1/relation.

4. Some element refinements violate the One-to-One Principle, according to which Dublin Core metadata describes (ideally) one manifestation or version of a resource, rather than conflating multiple manifestations into one description

(see Usage Guide [7], Section 1.2.). An example of this is:

Description.ProgramRelatedText

This is described as the actual text or link to text. This violates the One-to-One Principle inasmuch it is really another representation of the program in textual form -- a "related resource" and, as such, better referenced using a refinement of dc:relation.

5. Historically, the Dublin Core community has had difficulty distinguishing within the dc:type element between "form" and "genre". Rather than create two separate refinements of dc:type -- Type.Form and Type.Genre, as proposed in the draft PB Core -- it may be easiest simply to use two or more specialized vocabularies (one vocabulary of "forms" and one vocabulary of "genres") in conjunction with the unrefined element dc:type.

For presenting PB Core as a document, its authors might consider using the CEN guidelines for Dublin Core application profiles [9].

[1] http://dublincore.org/usage/meetings/2004/03/ISSUES/profiles-pbcore/
[2] http://dublincore.org/usage/documents/principles/
[3] http://dublincore.org/usage/documents/2003/11/18/principles/
[4] http://dublincore.org/documents/naming/
[5] http://dublincore.org/documents/dcmi-terms/
[6] http://www.ukoln.ac.uk/metadata/dcmi/abstract-model/2004-02-04/
[7] http://dublincore.org/documents/2002/09/24/library-application-profile/
[8] http://dublincore.org/documents/usageguide/
[9] http://www.cenorm.be/isss/cwa14855/

Title:	**Dublin Core Collection Description Application Profile Summary**
Creator:	Dublin Core Collection Description Working Group
Date Issued:	2005-03-19
Identifier:	http://www.ukoln.ac.uk/metadata/dcmi/collection-ap-summary/2005-03-19/
Replaces:	http://www.ukoln.ac.uk/metadata/dcmi/collection-ap-summary/2004-08-20/
Is Replaced By:	Not applicable
Latest Version:	http://www.ukoln.ac.uk/metadata/dcmi/collection-ap-summary/
Status of Document:	This is a DCMI Working Draft.
Description of Document:	This document presents a summary of the draft application profile for collection-level description developed by the Dublin Core Collection Description Working Group.

Note

Where names are listed below as Qualified Names, prefixes are assumed to be associated with Namespace Names as follows:

- **dc:** `http://purl.org/dc/elements/1.1/`
- **dcterms:** `http://purl.org/dc/terms/`
- **marcrel:** (MARC relator code properties) `http://www.loc.gov/marc.relators/`
- **cld:** (collection-specific properties: URIs to be confirmed, temporary Namespace Name/URI `http://example.org/cld/terms#`)
- **gen:** (non-collection-specific properties, URIs to be confirmed, temporary Namespace Name/URI `http://example.org/gen/terms#`)
- **cldtype:** (collection type terms: URIs to be confirmed, temporary Namespace Name/URI `http://example.org/cld/type#`)

Please note that where the Qualified Name of a term (property/class) appears against a shaded background in the table below (i.e. all terms with the prefixes `gen`, `cld` and `cldtype`), this indicates that those terms have not yet been assigned persistent URIs. Until such persistent URIs are assigned, by DCMI or by some other naming authority, these terms should be considered to be unstable and should not be referenced in metadata descriptions, except as part of the evaluation/testing of this profile.

Proposed collection properties

Attribute/ Property label	Property name (as Qualified Name)	Definition	Encoding Scheme(s)

Identifier	**dc:identifier**	A globally unique formal identifier for the collection.	dcterms:URI
Title	**dc:title**	The name of the collection.	
Alternative Title	**dcterms:alternative** (sub-property of dc:title)	Any form of name used as a substitute or alternative to the formal name of the collection	
Description	**dcterms:abstract** (sub-property of dc: description)	A summary description of the content of the collection.	
Physical Characteristics	**dc:format**	The physical or digital characteristics of the collection.	
Size	**dcterms:extent** (sub-property of dc:format)	The size of the collection.	
Language	**dc:language**	A language of the content of the items in the collection.	dcterms:ISO639-2
Type	**dc:type**	A type of the collection.	cld:CLDType
Rights	**dc:rights**	A statement of any rights held in/over the collection.	
Access Rights	**dcterms:accessRights** (sub-property of dc:rights)	A statement of any access restrictions placed on the collection, including allowed users, charges, etc.	
Accrual Method	**dcterms:accrualMethod**	The method by which items are added to the collection.	cld: DCCDAccrualMethod
Accrual Periodicity	**dcterms:accrualPeriodicity**	The frequency with which items are added to the collection.	cld: DCCDAccrualPeriodicity
Accrual Policy	**dcterms:accrualPolicy**	The policy governing the addition of items to the collection.	cld: DCCDAccrualPolicy
Custodial History	**dcterms:provenance**	A statement of any changes in ownership and custody of the collection that are significant for its authenticity, integrity and interpretation.	
Audience	**dcterms:audience**	A class of entity for whom the collection is intended or useful.	

Logo	**cld:logo** (sub-property of dc: description)	An image or logo associated with the collection.	
Subject	**dc:subject**	A subject or topic associated with the items in the collection.	dcterms:DDC dcterms:LCC dcterms:LCSH dcterms:MESH dcterms:UDC
Spatial Coverage	**dcterms:spatial** (sub-property of dc:coverage)	The spatial coverage of the content of the items in the collection.	
Temporal Coverage	**dcterms:temporal** (sub-property of dc:coverage)	The temporal coverage of the content of the items in the collection.	
Accumulation Date Range	**dcterms:created** (sub-property of dc:date)	The range of dates over which the collection was accumulated.	gen:ISO8601
Contents Date Range	**cld:dateContentsCreated** (sub-property of dc:date)	The range of dates over which the individual items within the collection were created	gen:ISO8601
Relationships between the Collection and Agents			
Collector	**dc:creator**	An entity who gathers (or gathered) the items in a collection together.	
Owner	**marcrel:own**	An entity who has legal possession of the collection.	
Relationships between the Collection and Location, Collection and Service			
Is Located At	**gen:isLocatedAt** (sub-property of dc:relation)	The location of the collection.	
Is Accessed Via	**gen:isAccessedVia** (sub-property of dc:relation)	A service that provides access to the collection.	
Relationships between Collections			
Sub-collection	**dcterms:hasPart** (sub-property of dc:relation)	A second collection contained within the current collection.	
Super-collection	**dcterms:isPartOf** (sub-property of dc:relation)	A second collection that contains the current collection.	

Catalogue or collection description	**dc:description**	A second collection that describes the current collection (for example, the catalogue for the current collection).	
Associated collection	**dc:relation**	A second collection that is associated with the current collection.	
Relationships between the Collection and other resources			
Associated publication	**dcterms:isReferencedBy** (sub-property of dc:relation)	A publication that is based on the use, study, or analysis of the collection.	

Changes made in this version

- For Accrual Method, used property dcterms:accrualMethod (rather than cld:accrualMethod) as term added to DCMI Namespace by Usage Board, October 2004.
- For Accrual Periodicity, used property dcterms:accrualPeriodicity (rather than cld:accrualPeriodicity) as term added to DCMI Namespace by Usage Board, October 2004.
- For Accrual Periodicity, used property dcterms:accrualPeriodicity (rather than cld:accrualPolicy) as term added to DCMI Namespace by Usage Board, October 2004.
- Altered name of date range encoding scheme from dcterms:ISO8601 to gen:ISO8601, as still under discussion by DC Date WG and DC Usage Board
- Altered name of Collection Type Vocabulary from cld:CLDT to cld:CLDType

---

Metadata associated with this resource: http://dublincore.org/documents/collection-ap-summary/index.shtml.rdf

Title:	**Dublin Core Collection Description Application Profile**
Creator:	Dublin Core Collection Description Working Group
Date Issued:	2005-03-19
Identifier:	http://www.ukoln.ac.uk/metadata/dcmi/collection-application-profile/2005-03-19/
Replaces:	http://www.ukoln.ac.uk/metadata/dcmi/collection-application-profile/2004-08-20/
Is Replaced By:	Not applicable
Latest Version:	http://www.ukoln.ac.uk/metadata/dcmi/collection-application-profile/
Status of Document:	This is a DCMI Working Draft.
Description of Document:	This document describes the draft application profile for collection-level description developed by the Dublin Core Collection Description Working Group.

Contents

Introduction

Collections and collection-level description

The term "collection" can be applied to any aggregation of physical or digital items. Those items may be of any type, so examples might include aggregations of natural objects, created objects, "born-digital" items, digital surrogates of physical items, and the catalogues of such collections (as aggregations of metadata records). The criteria for aggregation may vary: e.g. by location, by type or form of the items, by provenance of the items, by source or ownership, and so on. Collections may contain any number of items and may have varying levels of permanence.

A "collection-level description" provides a description of the collection as a unit: the resource described by a collection-level description is the collection, rather than the individual items within that collection.

Collection-level description enables a collection provider to

- **disclose** information about the content and availability of collections to users where item-level metadata does not exist or is not available, or where the provision of item-level detail is not required or appropriate

It enables a user to

- **discover** and **locate** collections of interest
- **select** collections to explore on the basis of a summary description
- **compare** collections as broadly similar objects, even where items (and/or item-level metadata) are heterogeneous
- understand conditions of **access** and **use**
- **interpret** collections (and items within collections)

Increasingly, many of these functions - notably the discovery, location, selection and comparison of collections - are being carried out by software acting on behalf of a human user, perhaps in accordance with user preferences or with parameters describing the scope of a particular service.

The Dublin Core Collection Description Application Profile (DC CD AP)

A DC application profile specifies which properties are used within a metadata application, and how those properties have been constrained or adapted for the purposes of that application. The DC CD AP specifies how properties from the Dublin Core metadata vocabularies (in association with properties from some other standard vocabularies) may be used for the creation of a collection-level description.

The DC CD AP is intended to provide a means of creating **simple** collection-level descriptions suitable for a **broad range** of collections. It is designed primarily to support the discovery and selection of collections, though it may be used to support other functions such as collection management too. It is not intended to describe every possible characteristic of every type of collection.

Also, it provides properties to describe the relationships between a collection and a number of other resources, notably the service that provides access to a collection, and some agents related to the collection. It is quite likely that an application will use metadata descriptions of those related resources as well as the collection, but the description of those resources is outside the scope of this document.

The DC CD AP is independent of any particular syntax for representing descriptions. Separate guidelines will describe how descriptions conforming to the DC CD AP may be represented using the conventions recommended by DCMI for expressing DC metadata using the Resource Description Framework (RDF) and using the Extensible Markup Language (XML).

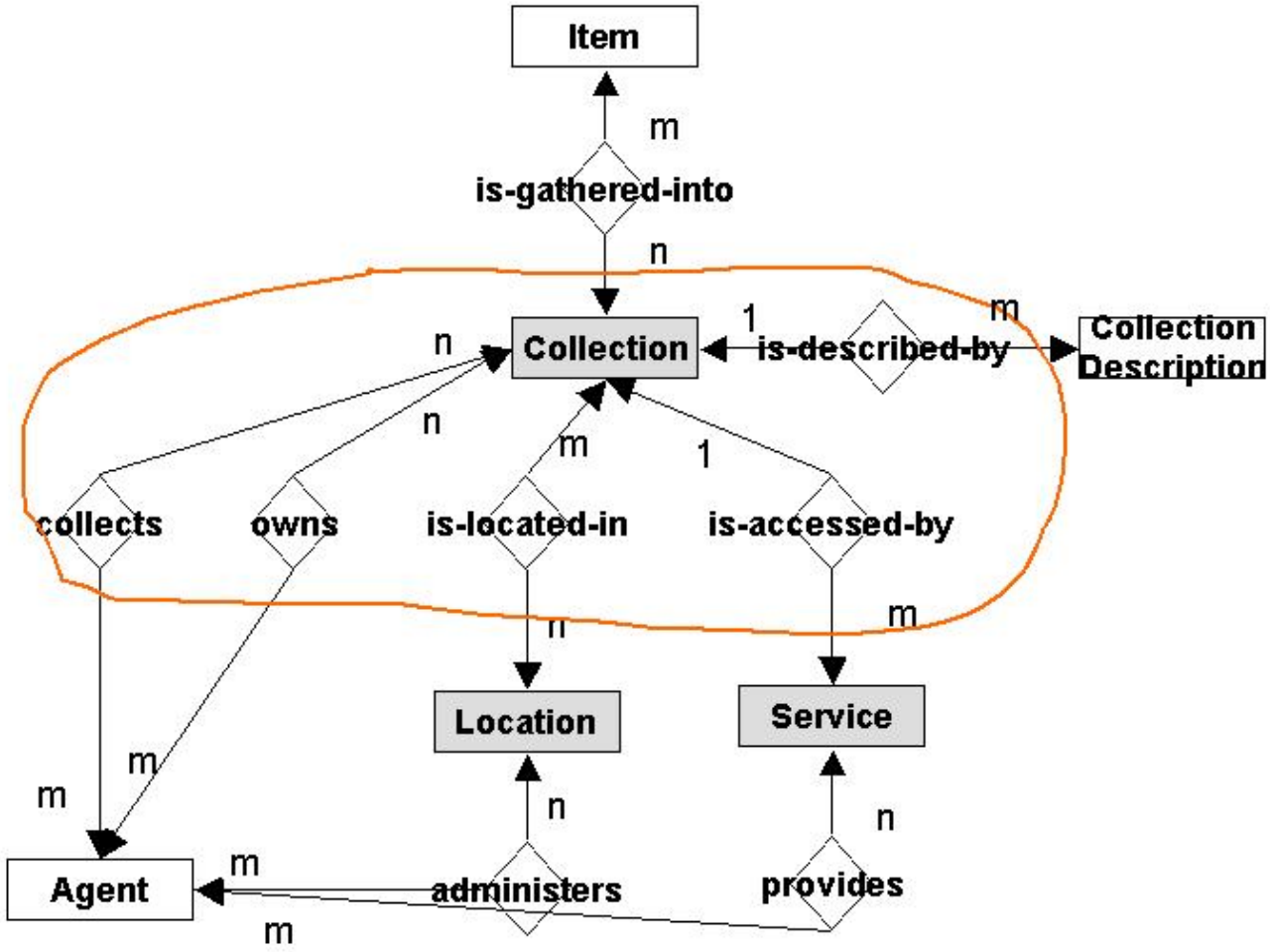
Functional Requirements of the DC CD AP

The aim is that the DC CD AP should support:

- the **discovery** of collections of potential interest using some common access points
- the **identification** of a known collection
- the **selection** of one or more collections from amongst a number of discovered collections
- the **identification** of the location of the collection
- the **identification** of the services that provide access to the collection

Data Model

The DC CD AP is based on a data model which is derived from that described in *An Analytical Model of Collections and their Catalogues* [1]. It differs from that model in adding a new entity type, Service.



In the Analytic Model relationships may carry attributes; in the DC CD AP, relationships are represented as simple properties and do not themselves carry attributes, so some of the expressivity of the model is lost in the metadata schema.

This version of the DC CD AP describes the use of properties to represent attributes of the collection and relationships between the collection and other entities. It does not describe how to represent properties of other entities in the model.

Vocabularies / Namespaces used

Vocabulary Name	URI	Prefix
The Dublin Core Metadata Element Set, v1.1	http://purl.org/dc/elements/1.1/	dc
Dublin Core Terms	http://purl.org/dc/terms/	dcterms
MARC Relator Codes	http://www.loc.gov/marc.relators/	marcrel
Collection Description Terms	http://example.org/cld/terms# [temporary URI, final URI to be confirmed]	cld
General Description Terms	http://example.org/gen/terms# [temporary URI, final URI to be confirmed]	gen
Collection Type Vocabulary Terms	http://example.org/cld/type# [temporary URI, final URI to be confirmed]	cldtype

Please note that where terms have Qualified Names with the prefixes `gen`, `cld` and `cldtype`, this indicates that those terms have not yet been assigned persistent URIs. Until such persistent URIs are assigned, by DCMI or by some other naming authority, these terms should be considered to be unstable and should not be referenced in metadata descriptions, except as part of the evaluation / testing of this profile.

Collection Properties

Shaded entries in the tables below indicate that either the semantics or the usage (e.g. labels, optionality, cardinality, use of Encoding Schemes) of a metadata element have been optimised/localised by this Application Profile.

- Collection Identifier [dc:identifier]
- Title [dc:title]
- Alternative Title [dcterms:alternative]
- Description [dcterms:abstract]
- Physical Characteristics [dc:format]
- Size [dcterms:extent]
- Language [dc:language]
- Type [dc:type]
- Rights [dc:rights]
- Access Rights [dcterms:accessRights]
- Accrual Method [dcterms:accrualMethod]
- Accrual Periodicity [dcterms:accrualPeriodicity]
- Accrual Policy [dcterms:accrualPolicy]
- Custodial History [dcterms:provenance]
- Audience [dcterms:audience]
- Logo [cld:logo]
- Subject [dc:subject]
- Spatial Coverage [dcterms:spatial]
- Temporal Coverage [dcterms:temporal]
- Accumulation Date Range [dcterms:created]
- Contents Date Range [cld:dateContentsCreated]
- Collector [dc:creator]
- Owner [marcrel:own]
- Is Located At [gen:isLocatedAt]
- Is Accessed Via [gen:isAccessedVia]
- Associated Publication [dcterms:isReferencedBy]
- Super-Collection [dcterms:isPartOf]
- Sub-Collection [dcterms:hasPart]
- Catalogue or Description [dc:description]
- Associated Collection [dc:relation]

Identifier [dc:identifier]

Identifier	**http://purl.org/dc/elements/1.1/identifier**
Name	identifier
Label	Collection Identifier
Defined By	The Dublin Core Metadata Element Set, v1.1 http://purl.org/dc/elements/1.1/
Source Definition	An unambiguous reference to the resource within a given context.
DC CD AP Definition	A globally unique formal identifier for the collection.
Comments	A collection identifier must be a URI, and the use of a URI scheme that has been registered with IANA is preferred.
Type of Term	Element
Refines	[n/a]
Refined by	[n/a]
Uses Encoding Scheme	Uniform Resource Identifier (URI), Dublin Core Terms [dcterms:URI] http://purl.org/dc/terms/URI
Similar to	[n/a]
Obligation	Optional, but recommended.
Condition	[n/a]
Datatype	[n/a]
Occurrence	Minimum: 0, Maximum: unbounded

Title [dc:title]

Identifier	**http://purl.org/dc/elements/1.1/title**
Name	title
Label	Title

Defined By	The Dublin Core Metadata Element Set, v1.1 http://purl.org/dc/elements/1.1/
Source Definition	A name given to the resource.
DC CD AP Definition	The name of the collection.
Comments	Enter the name of the collection preserving the original wording, order and spelling. Punctuation need not reflect the usage of the original. Subtitles should be separated from the title by a colon, for example: Voices from the Dust Bowl: The Charles L. Todd and Robert Sonkin Migrant Worker Collection
Type of Term	Element
Refines	[n/a]
Refined by	Alternative, Dublin Core Terms, [dcterms:alternative] http://purl.org/dc/terms/alternative
Uses Encoding Scheme	[n/a]
Similar to	[n/a]
Obligation	Mandatory
Condition	[n/a]
Datatype	String, with optional language identifier
Occurrence	Minimum: 1, Maximum: unbounded

Alternative Title [dcterms:alternative]

Identifier	**http://purl.org/dc/terms/alternative**
Name	alternative
Label	Alternative Title
Defined By	Dublin Core Terms http://purl.org/dc/terms/
Source Definition	Any form of the title used as a substitute or alternative to the formal title of the resource.
DC CD AP Definition	Any form of the name used as a substitute or alternative to the formal name of the collection.
Comments	Values may include acronyms that are used in addition to the name of the collection.
Type of Term	Element Refinement
Refines	Title, Dublin Core Metadata Element Set, v1.1 [dc:title] http://purl.org/dc/elements/1.1/title
Refined by	[n/a]
Uses Encoding Scheme	[n/a]
Similar to	[n/a]
Obligation	Optional
Condition	[n/a]
Datatype	String, with optional language identifier
Occurrence	Minimum: 0, Maximum: unbounded

Description [dcterms:abstract]

Identifier	**http://purl.org/dc/terms/abstract**
Name	abstract
Label	Description
Defined By	Dublin Core Terms http://purl.org/dc/terms/

Source Definition	A summary of the content of the resource.
DC CD AP Definition	A summary of the content of the collection.
Comments	A free text summary of the collection. Although a description might contain detailed subject-specific information, at least part of the description should be understandable by an end-user with no specialist knowledge of the subject area.
Type of Term	Element refinement
Refines	Description, Dublin Core Metadata Element Set, v1.1 [dc:description] http://purl.org/dc/elements/1.1/description
Uses Encoding Scheme	[n/a]
Similar to	[n/a]
Obligation	Mandatory
Condition	[n/a]
Datatype	String, with optional language identifier
Occurrence	Minimum: 1, Maximum: unbounded

Physical Characteristics [dc:format]

Identifier	**http://purl.org/dc/elements/1.1/format**
Name	format
Label	Physical Characteristics
Defined By	The Dublin Core Metadata Element Set, v1.1 http://purl.org/dc/elements/1.1/
Source Definition	The physical or digital manifestation of the resource.
DC CD AP Definition	The physical or digital characteristics of the collection.
Comments	[n/a]
Type of Term	Element
Refines	[n/a]
Refined by	Extent, Dublin Core Terms, [dcterms:extent] http://purl.org/dc/terms/extent
Uses Encoding Scheme	[n/a]
Similar to	[n/a]
Obligation	Optional
Condition	[n/a]
Datatype	String, with optional language identifier
Occurrence	Minimum: 0, Maximum: unbounded

Size [dcterms:extent]

Identifier	**http://purl.org/dc/terms/extent**
Name	extent
Label	Size
Defined By	Dublin Core Terms http://purl.org/dc/terms/
Source Definition	The size or duration of the resource.
DC CD AP Definition	The size of the collection.
Comments	[n/a]

Type of Term	Element Refinement
Refines	Format, Dublin Core Metadata Element Set, v1.1 [dc:format] http://purl.org/dc/elements/1.1/format
Refined by	[n/a]
Uses Encoding Scheme	[n/a]
Similar to	[n/a]
Obligation	Optional
Condition	[n/a]
Datatype	String, with optional language identifier
Occurrence	Minimum: 0, Maximum: unbounded

Language [dc:language]

Identifier	**http://purl.org/dc/elements/1.1/language**
Name	language
Label	Language
Defined By	The Dublin Core Metadata Element Set, v1.1 http://purl.org/dc/elements/1.1/
Source Definition	A language of the intellectual content of the resource.
DC CD AP Definition	A language of the content of the items in the collection.
Comments	Enter language names in the form of the three character code defined by ISO 639-2. Where more than one language is provided, a repeated attribute should be used for each.
Type of Term	Element
Refines	[n/a]
Refined by	[n/a]
Uses Encoding Scheme	ISO639-2, Dublin Core Terms [dcterms:ISO639-2] http://purl.org/dc/terms/ISO639-2
Similar to	[n/a]
Obligation	Optional
Condition	[n/a]
Datatype	[n/a]
Occurrence	Minimum: 0, Maximum: unbounded

Type [dc:type]

Identifier	**http://purl.org/dc/elements/1.1/type**
Name	type
Label	Type
Defined By	The Dublin Core Metadata Element Set, v1.1 http://purl.org/dc/elements/1.1/
Source Definition	The nature or genre of the content of the resource.
DC CD AP Definition	The type of the collection.
Comments	[n/a]
Type of Term	Element
Refines	[n/a]
Refined by	[n/a]
Uses Encoding Scheme	Collection Type Vocabulary, Collection Type Vocabulary [cld:CLDType] http://example.org/cld/terms#CLDType
Similar to	[n/a]

Obligation	Optional
Condition	[n/a]
Datatype	[n/a]
Occurrence	Minimum: 0, Maximum: unbounded

Rights [dc:rights]

Identifier	**http://purl.org/dc/elements/1.1/rights**
Name	rights
Label	Rights
Defined By	The Dublin Core Metadata Element Set, v1.1 http://purl.org/dc/elements/1.1/
Source Definition	Information about rights held in and over the resource.
DC CD AP Definition	A statement of any rights held in/over the collection.
Comments	[n/a]
Type of Term	Element
Refines	[n/a]
Refined by	Access Rights, Dublin Core Terms, [dcterms:accessRights] http://purl.org/dc/terms/accessRights
Uses Encoding Scheme	[n/a]
Similar to	[n/a]
Obligation	Optional
Condition	[n/a]
Datatype	String, with optional language identifier
Occurrence	Minimum: 0, Maximum: unbounded

Access Rights [dcterms:accessRights]

Identifier	**http://purl.org/dc/terms/accessRights**
Name	accessRights
Label	Access Rights
Defined By	Dublin Core Terms http://purl.org/dc/terms/
Source Definition	Information about who can access the resource or an indication of its security status.
DC CD AP Definition	A statement of any access restrictions placed on the collection, including allowed users, charges, etc.
Comments	[n/a]
Type of Term	Element Refinement
Refines	Rights, Dublin Core Metadata Element Set, v1.1 [dc:rights] http://purl.org/dc/elements/1.1/rights
Refined by	[n/a]
Uses Encoding Scheme	[n/a]
Similar to	[n/a]
Obligation	Optional
Condition	[n/a]
Datatype	String, with optional language identifier
Occurrence	Minimum: 0, Maximum: unbounded

Accrual Method [dcterms:accrualMethod]

Identifier	**http://purl.org/dc/terms/accrualMethod**

Name	accrualMethod
Label	Accrual Method
Defined By	Dublin Core Terms http://purl.org/dc/terms/
Source Definition	The method by which items are added to a collection.
Comments	Recommended best practice is to use a value from the DCCD Accrual Method encoding scheme.
Type of Term	Element
Refines	[n/a]
Refined by	[n/a]
Uses Encoding Scheme	DCCD Accrual Method, Collection Description Terms [cld:DCCDAccrualMethod] http://example.org/cld/terms#DCCDAccrualMethod
Similar to	[n/a]
Obligation	Optional
Condition	[n/a]
Datatype	[n/a]
Occurrence	Minimum: 0, Maximum: unbounded

Accrual Periodicity [dcterms:accrualPeriodicity]

Identifier	**http://purl.org/dc/terms/accrualPeriodicity**
Name	accrualPeriodicity
Label	Accrual Periodicity
Defined By	Dublin Core Terms http://purl.org/dc/terms/
Source Definition	The frequency with which items are added to a collection.
Comments	Recommended best practice is to use a value from the DCCD Accrual Periodicity encoding scheme.
Type of Term	Element
Refines	[n/a]
Refined by	[n/a]
Uses Encoding Scheme	DCCD Accrual Periodicity, Collection Description Terms [cld:DCCDAccrualPeriodicity] http://example.org/cld/terms#DCCDAccrualPeriodicity
Similar to	[n/a]
Obligation	Optional
Condition	[n/a]
Datatype	[n/a]
Occurrence	Minimum: 0, Maximum: unbounded

Accrual Policy [dcterms:accrualPolicy]

Identifier	**http://purl.org/dc/terms/accrualPolicy**
Name	accrualPolicy
Label	Accrual Policy
Defined By	Dublin Core Terms http://purl.org/dc/terms/
Source Definition	The policy governing the addition of items to a collection.
Comments	Recommended best practice is to use a value from the DCCD Accrual Policy encoding scheme.
Type of Term	Element
Refines	[n/a]
Refined by	[n/a]

Uses Encoding Scheme	DCCD Accrual Policy, Collection Description Terms [cld:DCCDAccrualPolicy] http://example.org/cld/terms#DCCDAccrualPolicy
Similar to	[n/a]
Obligation	Optional
Condition	[n/a]
Datatype	[n/a]
Occurrence	Minimum: 0, Maximum: unbounded

Custodial History [dcterms:provenance]

Identifier	**http://purl.org/dc/terms/provenance**
Name	provenance
Label	Custodial History
Defined By	Dublin Core Terms http://purl.org/dc/terms/
Source Definition	A statement of any changes in ownership and custody of the resource since its creation that are significant for its authenticity, integrity and interpretation.
DC CD AP Definition	A statement of any changes in ownership and custody of the collection that are significant for its authenticity, integrity and interpretation.
Comments	[n/a]
Type of Term	Element
Refines	[n/a]
Refined by	[n/a]
Uses Encoding Scheme	[n/a]
Similar to	[n/a]
Obligation	Optional
Condition	[n/a]
Datatype	String, with optional language identifier
Occurrence	Minimum: 0, Maximum: unbounded

Audience [dcterms:audience]

Identifier	**http://purl.org/dc/terms/audience**
Name	audience
Label	Audience
Defined By	Dublin Core Terms http://purl.org/dc/terms/
Source Definition	A class of entity for whom the resource is intended or useful.
DC CD AP Definition	A class of entity for whom the collection is intended or useful.
Comments	[n/a]
Type of Term	Element
Refines	[n/a]
Refined by	[n/a]
Uses Encoding Scheme	[n/a]
Similar to	[n/a]
Obligation	Optional
Condition	[n/a]
Datatype	String, with optional language identifier
Occurrence	Minimum: 0, Maximum: unbounded

Logo [cld:logo]

Identifier	**http://example.org/cld/terms#logo**
Name	logo
Label	Logo
Defined By	Collection Description Terms http://example.org/cld/terms#
Definition	An image or logo associated with the collection.
Comments	[n/a]
Type of Term	Element Refinement
Refines	Description, Dublin Core Metadata Element Set, v1.1 [dc:description] http://purl.org/dc/elements/1.1/description
Refined by	[n/a]
Uses Encoding Scheme	[n/a]
Similar to	[n/a]
Obligation	Optional
Condition	[n/a]
Datatype	
Occurrence	Minimum: 0, Maximum: unbounded

Subject [dc:subject]

Identifier	**http://purl.org/dc/elements/1.1/subject**
Name	subject
Label	Subject
Defined By	The Dublin Core Metadata Element Set, v1.1 http://purl.org/dc/elements/1.1/
Source Definition	The topic of the content of the resource.
DC CD AP Definition	A subject or topic associated with the items in the collection.
Comments	Keywords or subject descriptors associated with items in the collection. The terms used indicate the subject matter of the collection. The encoding schemes below are those recommended by DCMI. Other appropriate encoding schemes may be used, and the use of a scheme should be indicated.
Type of Term	Element
Refines	[n/a]
Refined by	[n/a]
Uses Encoding Scheme	Library of Congress Subject Headings, Dublin Core Terms [dcterms:LCSH] http://purl.org/dc/terms/LCSH
Uses Encoding Scheme	Library of Congress Classification, Dublin Core Terms [dcterms:LCC] http://purl.org/dc/terms/LCC
Uses Encoding Scheme	Medical Subject Headings, Dublin Core Terms [dcterms:MESH] http://purl.org/dc/terms/MESH
Uses Encoding Scheme	Dewey Decimal Classification, Dublin Core Terms [dcterms:DDC] http://purl.org/dc/terms/DDC
Uses Encoding Scheme	Universal Decimal Classification, Dublin Core Terms [dcterms:UDC] http://purl.org/dc/terms/UDC
Similar to	[n/a]
Obligation	Optional
Condition	[n/a]
Datatype	[n/a]
Occurrence	Minimum: 0, Maximum: unbounded

Spatial Coverage [dcterms:spatial]

Identifier	**http://purl.org/dc/terms/spatial**
Name	spatial
Label	Spatial Coverage
Defined By	Dublin Core Terms http://purl.org/dc/terms/
Source Definition	Spatial characteristics of the intellectual content of the resource.
DC CD AP Definition	The spatial coverage of the content of the items in the collection.
Comments	[n/a]
Type of Term	Element Refinement
Refines	Coverage, Dublin Core Metadata Element Set, v1.1 [dc:coverage] http://purl.org/dc/elements/1.1/coverage
Refined by	[n/a]
Uses Encoding Scheme	[n/a]
Similar to	[n/a]
Obligation	Optional
Condition	[n/a]
Datatype	[n/a]
Occurrence	Minimum: 0, Maximum: unbounded

Temporal Coverage [dcterms:temporal]

Identifier	**http://purl.org/dc/terms/temporal**
Name	temporal
Label	Temporal Coverage
Defined By	Dublin Core Terms http://purl.org/dc/terms/
Source Definition	Temporal characteristics of the intellectual content of the resource.
DC CD AP Definition	The temporal coverage of the content of the items in the collection.
Comments	[n/a]
Type of Term	Element Refinement
Refines	Coverage, Dublin Core Metadata Element Set, v1.1 [dc:coverage] http://purl.org/dc/elements/1.1/coverage
Refined by	[n/a]
Uses Encoding Scheme	[n/a]
Similar to	[n/a]
Obligation	Optional
Condition	[n/a]
Datatype	String
Occurrence	Minimum: 0, Maximum: unbounded

Accumulation Date Range [dcterms:created]

Identifier	**http://purl.org/dc/terms/created**
Name	created
Label	Accumulation Date Range
Defined By	Dublin Core Terms http://purl.org/dc/terms/
Source Definition	Date of creation of the resource.

DC CD AP Definition	The range of dates over which the collection was accumulated.
Comments	A date range should be supplied in the form of a time interval conforming to ISO8601. Start and end dates should be separated by a forward-slash (/). Each date should be entered according to the W3C note on 'Date and Time Formats', <http://www.w3.org/TR/NOTE-datetime>. Null dates may be used to indicate open-ended date ranges. All the following are examples of valid date ranges: 1888/1894 1960/ 2000-02/2000-06 indicating, '1888 to 1894 inclusive', '1960 onwards' and 'February 2000 to June 2000 inclusive' respectively.
Type of Term	Element Refinement
Refines	Date, Dublin Core Metadata Element Set, v1.1 [dc:date] http://purl.org/dc/elements/1.1/date
Refined by	[n/a]
Uses Encoding Scheme	ISO8601, General Description Terms [gen:ISO8601] http://example.org/gen/terms#ISO8601
Similar to	[n/a]
Obligation	Optional
Condition	[n/a]
Datatype	String
Occurrence	Minimum: 0, Maximum: unbounded

Contents Date Range [cld:dateContentsCreated]

Identifier	**http://example.org/cld/terms#dateContentsCreated**
Name	dateContentsCreated
Label	Contents Date Range
Defined By	Collection Description Terms http://example.org/cld/terms#
Definition	The range of dates over which the individual items within the collection were created.
Comments	A date range should be supplied in the form of a time interval conforming to ISO8601. Start and end dates should be separated by a forward-slash (/). Each date should be entered according to the W3C note on 'Date and Time Formats', <http://www.w3.org/TR/NOTE-datetime>. Null dates may be used to indicate open-ended date ranges. All the following are examples of valid date ranges: 1888/1894 1960/ 2000-02/2000-06 indicating, '1888 to 1894 inclusive', '1960 onwards' and 'February 2000 to June 2000 inclusive' respectively.
Type of Term	Element Refinement
Refines	Date, Dublin Core Metadata Element Set, v1.1 [dc:date] http://purl.org/dc/elements/1.1/date
Refined by	[n/a]
Uses Encoding Scheme	ISO8601, General Description Terms [gen:ISO8601] http://example.org/gen/terms#ISO8601
Similar to	[n/a]
Obligation	Optional
Condition	[n/a]

Datatype	String
Occurrence	Minimum: 0, Maximum: unbounded

Collector [dc:creator]

Identifier	**http://purl.org/dc/elements/1.1/creator**
Name	creator
Label	Collector
Defined By	The Dublin Core Metadata Element Set, v1.1 http://purl.org/dc/elements/1.1/
Source Definition	An entity primarily responsible for making the content of the resource.
DC CD AP Definition	An entity who gathers (or gathered) the items in a collection together.
Comments	[n/a]
Type of Term	Element
Refines	[n/a]
Refined by	[n/a]
Uses Encoding Scheme	[n/a]
Similar to	[n/a]
Obligation	Optional
Condition	[n/a]
Datatype	
Occurrence	Minimum: 0, Maximum: unbounded

Owner [marcrel:own]

Identifier	**http://www.loc.gov/marc.relators/own**
Name	own
Label	Owner
Defined By	MARC Relator Codes http://www.loc.gov/marc.relators/
Source Definition	The person or organization that currently owns an item or collection.
DC CD AP Definition	An entity who has legal possession of the collection.
Comments	[n/a]
Type of Term	Element
Refines	[n/a]
Refined by	[n/a]
Uses Encoding Scheme	[n/a]
Similar to	[n/a]
Obligation	Optional
Condition	[n/a]
Datatype	
Occurrence	Minimum: 0, Maximum: unbounded

Is Located At [gen:isLocatedAt]

Identifier	**http://example.org/gen/terms#isLocatedAt**
Name	isLocatedAt
Label	Is Located At
Defined By	General Description Terms http://example.org/gen/terms#

Definition	A location of the collection.
Comments	
Type of Term	Element Refinement
Refines	Relation, Dublin Core Metadata Element Set, v1.1 [dc:relation] http://purl.org/dc/elements/1.1/relation
Refined by	[n/a]
Uses Encoding Scheme	[n/a]
Similar to	[n/a]
Obligation	Optional
Condition	[n/a]
Datatype	String
Occurrence	Minimum: 0, Maximum: unbounded

Is Accessed Via [gen:isAccessedVia]

Identifier	**http://example.org/gen/terms#isAccessedVia**
Name	isAccessedVia
Label	Is Accessed Via
Defined By	General Description Terms http://example.org/gen/terms#
Definition	A service that provides access to the collection.
Comments	
Type of Term	Element Refinement
Refines	Relation, Dublin Core Metadata Element Set, v1.1 [dc:relation] http://purl.org/dc/elements/1.1/relation
Refined by	[n/a]
Uses Encoding Scheme	[n/a]
Similar to	[n/a]
Obligation	Optional
Condition	[n/a]
Datatype	String
Occurrence	Minimum: 0, Maximum: unbounded

Sub-collection [dcterms:hasPart]

Identifier	**http://purl.org/dc/terms/hasPart**
Name	hasPart
Label	Sub-collection
Defined By	Dublin Core Terms http://purl.org/dc/terms/
Source Definition	The described resource includes the referenced resource either physically or logically.
DC CD AP Definition	A second collection contained within the current collection.
Comments	[n/a]
Type of Term	Element Refinement
Refines	Relation, Dublin Core Metadata Element Set, v1.1 [dc:relation] http://purl.org/dc/elements/1.1/relation
Refined by	[n/a]
Uses Encoding Scheme	[n/a]
Similar to	[n/a]
Obligation	Optional
Condition	[n/a]

Datatype	
Occurrence	Minimum: 0, Maximum: unbounded

Super-collection [dcterms:isPartOf]

Identifier	**http://purl.org/dc/terms/isPartOf**
Name	isPartOf
Label	Super-collection
Defined By	Dublin Core Terms http://purl.org/dc/terms/
Source Definition	The described resource is a physical or logical part of the referenced resource.
DC CD AP Definition	A second collection that contains the current collection.
Comments	[n/a]
Type of Term	Element Refinement
Refines	Relation, Dublin Core Metadata Element Set, v1.1 [dc:relation] http://purl.org/dc/elements/1.1/relation
Refined by	[n/a]
Uses Encoding Scheme	[n/a]
Similar to	[n/a]
Obligation	Optional
Condition	[n/a]
Datatype	
Occurrence	Minimum: 0, Maximum: unbounded

Catalogue or description [dc:description]

Identifier	**http://purl.org/dc/elements/1.1/description**
Name	description
Label	Catalogue or description
Defined By	The Dublin Core Metadata Element Set, v1.1 http://purl.org/dc/elements/1.1/
Source Definition	An account of the content of the resource.
DC CD AP Definition	A second collection that describes the current collection (for example, the catalogue for the current collection).
Comments	[n/a]
Type of Term	Element
Refines	[n/a]
Refined by	Description, Dublin Core Terms, [dcterms:abstract] http://purl.org/dc/terms/abstract
Uses Encoding Scheme	[n/a]
Similar to	[n/a]
Obligation	Optional
Condition	[n/a]
Datatype	
Occurrence	Minimum: 0, Maximum: unbounded

Associated collection [dc:relation]

Identifier	**http://purl.org/dc/elements/1.1/relation**
Name	relation
Label	Associated collection

Defined By	The Dublin Core Metadata Element Set, v1.1 http://purl.org/dc/elements/1.1/
Source Definition	A reference to a related resource.
DC CD AP Definition	A second collection that is associated with the current collection.
Comments	[n/a]
Type of Term	Element Refinement
Refines	[n/a]
Refined by	[n/a]
Uses Encoding Scheme	[n/a]
Similar to	[n/a]
Obligation	Optional
Condition	[n/a]
Datatype	
Occurrence	Minimum: 0, Maximum: unbounded

Associated publication [dcterms:isReferencedBy]

Identifier	**http://purl.org/dc/terms/isReferencedBy**
Name	isReferencedBy
Label	Associated publication
Defined By	Dublin Core Terms http://purl.org/dc/terms/
Source Definition	The described resource is referenced, cited, or otherwise pointed to by the referenced resource.
DC CD AP Definition	A publication that is based on the use, study, or analysis of the collection.
Comments	[n/a]
Type of Term	Element Refinement
Refines	Relation, Dublin Core Metadata Element Set, v1.1 [dc:relation] http://purl.org/dc/elements/1.1/relation
Refined by	[n/a]
Uses Encoding Scheme	[n/a]
Similar to	[n/a]
Obligation	Optional
Condition	[n/a]
Datatype	
Occurrence	Minimum: 0, Maximum: unbounded

Encoding Schemes Used

- Uniform Resource Identifier [dcterms:URI]
- ISO 639-2 [dcterms:ISO639-2]
- DCCD Accrual Method [cld:DCCDAccrualMethod]
- DCCD Accrual Periodicity [cld:DCCDAccrualPeriodicity]
- DCCD Accrual Policy [cld:DCCDAccrualPolicy]
- Library of Congress Subject Headings [dcterms:LCSH]
- Library of Congress Classification [dcterms:LCC]
- Medical Subject Headings [dcterms:MESH]
- Dewey Decimal Classification [dcterms:DDC]
- Universal Decimal Classification [dcterms:UDC]
- W3C Date and Time Format [dcterms:W3CDTF]
- ISO 8601 [gen:ISO8601]
- Collection Type Vocabulary [cld:CLDType]

Uniform Resource Identifier [dcterms:URI]

Identifier	**http://purl.org/dc/terms/URI**
Name	URI
Label	Uniform Resource Identifier
Defined By	Dublin Core Terms http://purl.org/dc/terms/
Definition	Uniform Resource Identifier.
Comments	[n/a]
See Also	http://www.ietf.org/rfc/rfc2396.txt
Type of Term	Encoding Scheme
Encoding Scheme for	Identifier, Dublin Core Metadata Element Set, v1.1 [dc:identifier] http://purl.org/dc/elements/1.1/identifier

ISO639-2 [dcterms:ISO639-2]

Identifier	**http://purl.org/dc/terms/ISO639-2**
Name	ISO639-2
Label	ISO 639-2
Defined By	Dublin Core Terms http://purl.org/dc/terms/
Definition	ISO 639-2: Codes for the representation of names of languages.
Comments	[n/a]
See Also	http://www.loc.gov/standards/iso639-2/
Type of Term	Encoding Scheme
Encoding Scheme for	Language, Dublin Core Metadata Element Set, v1.1 [dc:language] http://purl.org/dc/elements/1.1/language

Collection Type Vocabulary [cld:CLDType]

Identifier	**http://example.org/cld/terms#CLDType**
Name	CLDType
Label	Collection Type Vocabulary
Defined By	Collection Description Terms http://example.org/cld/terms#
Definition	A list of types that categorize the collection according to the nature of the items in the collection.
Comments	[n/a]
See Also	http://www.ukoln.ac.uk/metadata/dcmi/collection-type/
Type of Term	Encoding Scheme
Encoding Scheme for	Type, Dublin Core Metadata Element Set, v1.1 [dc:type] http://purl.org/dc/elements/1.1/type

DCCD Accrual Method [cld:DCCDAccrualMethod]

Identifier	**http://example.org/cld/terms#DCCDAccrualMethod**
Name	DCCDAccrualMethod
Label	DCCD Accrual Method
Defined By	Collection Description Terms http://example.org/cld/terms#
Definition	Methods by which items are added to a collection.
Comments	[n/a]
See Also	http://www.ukoln.ac.uk/metadata/dcmi/collection-DCCDAccrualMethod/
Type of Term	Encoding Scheme

Encoding Scheme for	Accrual Method, Dublin Core Terms [dcterms:accrualMethod] http://purl.org/dc/terms/accrualMethod

DCCD Accrual Periodicity [cld:DCCDAccrualPeriodicity]

Identifier	**http://example.org/cld/terms#DCCDAccrualPeriodicity**
Name	DCCDAccrualPeriodicity
Label	DCCD Accrual Periodicity
Defined By	Collection Description Terms http://example.org/cld/terms#
Definition	Frequencies which items are added to a collection.
Comments	[n/a]
See Also	http://www.ukoln.ac.uk/metadata/dcmi/collection-DCCDAccrualPeriodicity/
Type of Term	Encoding Scheme
Encoding Scheme for	Accrual Periodicity, Dublin Core Terms [dcterms:accrualPeriodicity] http://purl.org/dc/terms/accrualPeriodicity

DCCD Accrual Policy [cld:DCCDAccrualPolicy]

Identifier	**http://example.org/cld/terms#DCCDAccrualPolicy**
Name	DCCDAccrualPolicy
Label	DCCD Accrual Policy
Defined By	Collection Description Terms http://example.org/cld/terms#
Definition	Policies governing the addition of items to a collection.
Comments	[n/a]
See Also	http://www.ukoln.ac.uk/metadata/dcmi/collection-DCCDAccrualPolicy/
Type of Term	Encoding Scheme
Encoding Scheme for	Accrual Policy, Dublin Core Terms [dcterms:accrualPolicy] http://purl.org/dc/terms/accrualPolicy

Library of Congress Subject Headings (LCSH) [dcterms:LCSH]

Identifier	**http://purl.org/dc/terms/LCSH**
Name	LCSH
Label	LCSH
Defined By	Dublin Core Terms http://purl.org/dc/terms/
Definition	Library of Congress Subject Headings.
Comments	[n/a]
See Also	http://lcweb.loc.gov/cds/lcsh.html
Type of Term	Encoding Scheme
Encoding Scheme for	Subject, Dublin Core Metadata Element Set, v1.1 [dc:subject] http://purl.org/dc/elements/1.1/subject

Library of Congress Classification (LCC) [dcterms:LCC]

Identifier	**http://purl.org/dc/terms/LCC**
Name	LCC
Label	LCC
Defined By	Dublin Core Terms http://purl.org/dc/terms/

Definition	Library of Congress Classification.
Comments	[n/a]
See Also	http://lcweb.loc.gov/catdir/cpso/lcco/lcco.html
Type of Term	Encoding Scheme
Encoding Scheme for	Subject, Dublin Core Metadata Element Set, v1.1 [dc:subject] http://purl.org/dc/elements/1.1/subject

Medical Subject Headings (MESH) [dcterms:MESH]

Identifier	**http://purl.org/dc/terms/MESH**
Name	MESH
Label	MeSH
Defined By	Dublin Core Terms http://purl.org/dc/terms/
Definition	Medical Subject Headings.
Comments	[n/a]
See Also	http://www.nlm.nih.gov/mesh/meshhome.html
Type of Term	Encoding Scheme
Encoding Scheme for	Subject, Dublin Core Metadata Element Set, v1.1 [dc:subject] http://purl.org/dc/elements/1.1/subject

Dewey Decimal Classification (DDC) [dcterms:DDC]

Identifier	**http://purl.org/dc/terms/DDC**
Name	DDC
Label	DDC
Defined By	Dublin Core Terms http://purl.org/dc/terms/
Definition	Dewey Decimal Classification.
Comments	[n/a]
See Also	http://www.oclc.org/dewey/
Type of Term	Encoding Scheme
Encoding Scheme for	Subject, Dublin Core Metadata Element Set, v1.1 [dc:subject] http://purl.org/dc/elements/1.1/subject

Universal Decimal Classification (UDC) [dcterms:UDC]

Identifier	**http://purl.org/dc/terms/UDC**
Name	UDC
Label	UDC
Defined By	Dublin Core Terms http://purl.org/dc/terms/
Definition	Universal Decimal Classification.
Comments	[n/a]
See Also	http://www.udcc.org/
Type of Term	Encoding Scheme
Encoding Scheme for	Subject, Dublin Core Metadata Element Set, v1.1 [dc:subject] http://purl.org/dc/elements/1.1/subject

W3C Date and Time Format (W3CDTF) [dcterms:W3CDTF]

Identifier	**http://purl.org/dc/terms/W3CDTF**
Name	W3CDTF
Label	W3C-DTF
Defined By	Dublin Core Terms http://purl.org/dc/terms/
Definition	W3C Encoding rules for dates and times - a profile based on ISO 8601.
Comments	[n/a]
See Also	http://www.w3.org/TR/NOTE-datetime
Type of Term	Encoding Scheme
Encoding Scheme for	

ISO 8601 [gen:ISO8601]

Identifier	**http://example.org/gen/terms#ISO8601**
Name	ISO8601
Label	ISO 8601
Defined By	General Description Terms http://example.org/gen/terms#
Definition	A date range - two dates separated by a forward-slash (/). Each date should be entered according to the W3C note on 'Date and Time Formats'. Null dates may be used to indicate open-ended date ranges.
Comments	[n/a]
See Also	[n/a]
Type of Term	Encoding Scheme
Encoding Scheme for	Contents Date Range, Collection Description Terms, [cld:contentsDateRange] http://example.org/cld/terms/contentsDateRange
Encoding Scheme for	Accumulation Date Range, Dublin Core Terms, [dcterms:created] http://purl.org/dc/terms/created

Administrative Metadata

A collection-level description conforming to this profile **should** itself be described by an appropriate administrative metadata record. This document does not specify the properties to be used in that administrative metadata record.

Changes made in this version

- For Accrual Method, used property dcterms:accrualMethod (rather than cld:accrualMethod) as term added to DCMI Namespace by Usage Board, October 2004.
- For Accrual Periodicity, used property dcterms:accrualPeriodicity (rather than cld:accrualPeriodicity) as term added to DCMI Namespace by Usage Board, October 2004.
- For Accrual Periodicity, used property dcterms:accrualPeriodicity (rather than cld:accrualPolicy) as term added to DCMI Namespace by Usage Board, October 2004.
- Altered name of date range encoding scheme from dcterms:ISO8601 to gen:ISO8601, as still under discussion by DC Date WG and DC Usage Board
- Altered name of Collection Type Vocabulary from cld:CLDT to cld:CLDType
- Replaced name of property Is Available Via (gen:isAvailableVia) with Is Accessed Via (gen:isAccessedVia).
- Added text for "source definitions".

References

[1] Michael Heaney. *An Analytical Model of Collections and their Catalogues*
http://www.ukoln.ac.uk/metadata/rslp/model/

Metadata associated with this resource: http://dublincore.org/documents/collection-application-profile/index.shtml.rdf

Current Issues with DC Collection Description Application Profile, May 2005

Pete Johnston, 2005-05-11

1. Clarifying the distinction between the isLocatedAt (Collection-Location) and isAccessedVia (Collection-Service) relationships

Mike Heaney (who developed the Entity-Relation Model on which the RSLP CD schema and the DC CD AP is based) is currently working on an extension to the model to address this. We (some UKOLN-folk) have seen a draft, and the final document is due within the next couple of weeks. This document will be circulated to the DC CD WG and the definitions of the properties may be amended as a result.

2. One-to-One rule concerns

It was suggested that a functional requirement for the DC CD AP was that it should support the discovery of collections according to the media-type of the items within the collection e.g. a collection containing (a significant proportion of) items of mp3 format etc.

The WG has not resolved how to express this information, within the constraints of the one-to-one rule (the format of an item is not the format of acollection). And there have been some suggestions that it may not be a requirement.

This issue is currently under discussion on the WG mailing list [1].

3. Date Formats

Finalising the DC CD AP depends on DC Date WG and/or Usage Board finalising work on date encoding schemes which support date ranges and approximate dates. The DC CD WG would welcome an update on progress.

4. Identifiers for terms

The development of the DC CD AP has resulted in the creation of several new metadata terms not currently available in other vocabularies. Although (in most cases, at least), the WG has agreed on the definitions of these terms, they have not yet been assigned persistent URIs because a suitable naming authority and/or maintenanace authority has not been found.

A proposal has been made that the DC CD WG, or an agency acting on its behalf, should obtain a PURL top-level domain and assign some suitable PURLs for these terms now, with the intention of seeking an organisation to take over the longer term maintenanace of the terms [2].

[1] http://www.jiscmail.ac.uk/cgi-bin/webadmin?A2=ind0505&L=dc-collections&T=0&F=&S=&P=55
[2] http://www.jiscmail.ac.uk/cgi-bin/webadmin?A2=ind0504&L=dc-collections&T=0&F=&S=&P=562

Title: Decision on proposal for a Collection Description profile
Shepherd: Andrew Wilson
Identifier: http://dublincore.org/usage/decisions/2004/2004-02.Collection-terms.shtml
Date: 2004-09-03
Description: The decisions documented here refer to proposals considered at the Usage Board meeting of March 2004 in Bath UK.

Text of proposals:
-- http://www.ukoln.ac.uk/metadata/dcmi/collection-provenance/2004-02-10/
-- http://www.ukoln.ac.uk/metadata/dcmi/collection-isAvailableAt/2004-01-24/

Decision: The Usage Board approves the addition of a new element -- "Provenance" -- as a Conforming term in the dcterms namespace. The Usage Board does not approve the proposed new element refinement "isAvailableAt".

Discussion

The Collection Description Working Group proposed the addition of two new terms: "provenance" as a refinement of dc:description; and "isAvailableAt" as a refinement of dc:relation. The DCMI Usage Board reviewed the proposal at a meeting in Bath UK on Sunday, 14 March 2004. Members present were Tom Baker (chair), Diane Hillmann, Akira Miyazawa, Andy Powell, Roland Schwaenzl, Stuart Sutton, Rebecca Guenther, and Andrew Wilson (designated shepherd of the Collection Description terms proposal).

Discussion of "provenance" centred around the definition, and whether the proposal of "provenance" as a refinement of dc:description was appropriate. The UB agreed on a revised definition of "provenance" with additional information in the comment field of the proposal text. UB decided that "provenance" had wider resource discovery application than just within the collection description domain and agreed to approve "provenance" as a new conforming element (property) in its own right in the dcterms namespace.

In the UB discussion of the proposed refinement "isAvailableAt" -- both at the Bath meeting and subsequently on the mailing list -- the following points were made:

-- The Collection Description working group would like to distinguish between an "identifier" for a resource (i.e., a string designating the resource described) and a "locator" usable for accessing that resource.

-- The Collection Description working group also would like to be able to describe a service which "provides access to" that resource as an entity in its own right -- with, in principle, its own set of attributes (i.e., as a "related resource" related to the resource described). This was the rationale for proposing "isAvailableAt" as a refinement of dc:relation.

-- Metadata aggregators such as NSDL and AGLS find that, in practice, the value of dc:identifier is very commonly not an "identifier" in the purest sense of the word (i.e., a unique string not necessarily resolvable to a Web address), but rather a URL by which the resource can be accessed (i.e., a "locator"). In doing so, metadata authors are merely reflecting the endemic ambiguity between "identification" and "location" in the context of the Web.

-- Although the intention of the proposers of "Is

Available At" was to point to "a service"
making the resource available, it was feared that
dct:isAvailableAt might be used for the "locator" of
the resource itself. Such usage would merely compound
the ambiguity already surrounding dc:identifier with
a new ambiguity with respect to dct:isAvailableAt.

-- Specifically, it was feared that if metadata authors
were to put the locators of resources into a refinement
of dc:relation, then the fact that those URIs were
locators of the resource would be lost in the process
of dumbing down. After dumb-down, an aggregator might
be left with multiple values for dc:relation and have
no way of knowing or inferring which ones were usable
as locators for the resource.

-- It was pointed out that the proposed definition of
"Is Available At" ("The referenced resource provides
access to the described resource") is difficult to
distinguish in its intent from the definition of the
existing element dc:publisher ("An entity responsible
for making the resource available").

In sum, future proposals addressing these issues should
consider the following:

-- As argued by the Collection Description Working Group,
it may be desirable to distinguish more cleanly
between identifiers and locators for the resource
described. Any proposal to do so, however, should
address the ambiguity inherent in the existing usage
of dc:identifier.

-- It may be desirable to have a property specifically for
information services so that those services can be pointed
to or described as "related resources" -- i.e., as entities
in their own right.

For the practical purposes of aggregators, however,
it is not desirable that locators for those services be
associated with properties that are subject to dumb-down
to very broad and generic properties such as dc:relation.

Rather, information about the service should be provided in
some other manner. This information could be provided by a
a new top-level DCMI element, by using an existing property
from another namespace, or possibly by dc:publisher.

Future proposals should also be aware that elements beyond
the fifteen elements of DCMES 1.1 are likely not to be quickly
adopted due to the widespread use of "Simple Dublin Core" as
the shared schema of many content federations, and therefore as
the target of dumb-down. Proposals should, therefore, consider
the possibility that suggestions for putting significant
location information somewhere other than in one of the
original DC-15 elements may suffer from the risk that users
who (for whatever reason) limit their view to the DC-15 will
not see that location information.

There was some discussion about the MODS element for location,
which arguably has the same function as "isAvailableAt". The
issue here is the general one of the re-use of
properties that already exist in other namespaces.
This discussion led on to a broader consideration of the
difference between an XML element and an RDF property,
and whether the inherent differences (in modeling terms)
between XML elements and RDF properties means that XML
elements should be recommended for reuse as RDF properties

only under certain conditions. The Usage Board agreed there was a need to develop and write up a policy on XML elements and RDF properties to be discussed in DCMI at a later date.

Approved text - beginning

VMS-ID: provenance-001
Name: provenance
URI: http://purl.org/dc/terms/provenance
Namespace: http://purl.org/dc/terms/
Label: Provenance
Definition: A statement of any changes in ownership and custody of the resource since its creation that are significant for its authenticity, integrity and interpretation.
Comment: The statement may include a description of any changes successive custodians made to the resource.
Type of term: http://dublincore.org/usage/documents/principles/#element
Status: http://dublincore.org/usage/documents/process/#conforming
Date issued: 2004-09-20
Decision: http://dublincore.org/usage/decisions/#Decision-2004-02

Approved text - end

Title: Decision on a proposal for new terms for describing collections of resources
Shepherd: Andrew Wilson
Identifier: http://dublincore.org/usage/decisions/2005/2005-02.Collection-terms.shtml
Date: 2005-04-21
Description: The decisions documented here refer to proposals considered at the Usage Board meeting of October 2004 in Shanghai, China.

Text of proposals:
-- Accrual Method (element)
http://www.ukoln.ac.uk/metadata/dcmi/collection-accrualMethod/
-- Accrual Periodicity (element)
http://www.ukoln.ac.uk/metadata/dcmi/collection-accrualPeriodicity/
-- Accrual Policy (element)
http://www.ukoln.ac.uk/metadata/dcmi/collection-accrualPolicy/
-- Dublin Core Collection Description Accrual Method (encoding scheme)
http://www.ukoln.ac.uk/metadata/dcmi/collection-DCCDAccrualMethod/
-- Dublin Core Collection Description Accrual Periodicity (encoding scheme)
http://www.ukoln.ac.uk/metadata/dcmi/collection-DCCDAccrualPeriodicity/
-- Dublin Core Collection Description Accrual Policy (encoding scheme)
http://www.ukoln.ac.uk/metadata/dcmi/collection-DCCDAccrualPolicy/

Decision: The Usage Board approves the addition of three new terms -- "accrualMethod", "accrualPeriodicity" and "accrualPolicy" -- as conforming terms in the dcterms namespace. The Usage Board deferred a decision on the three proposed encoding schemes pending clarification of the ongoing maintenance of the vocabularies.

Discussion

The Collection Description Working Group proposed the addition of three new elements: "accrualMethod", "accrualPeriodicity", and "accrualPolicy"; and three new encoding schemes: "DCCDAccrualMethod", "DCCDAccrualPeriodicity", and "DCCDAccrualPolicy", as part of its development of an application profile for describing collections of resources. The DCMI Usage Board reviewed the proposal at a meeting in Shanghai China on Sunday, 10 October 2004. Members present were Tom Baker(chair), Diane Hillmann, Akira Miyazawa, Andy Powell, Stuart Sutton, Rebecca Guenther, and Andrew Wilson (designated shepherd of the Collection Description terms proposal).

Discussion of the three elements was concerned largely with the nature of "resource discovery" and whether the proposals actually achieved the aims of DCMI for cross-domain resource discovery. As a result of this discussion the Usage Board will clarify what it actually means when it refers to "resource discovery" and "cross-domain".

The three proposed elements and the three proposed encoding schemes were considered together since the encoding schemes were recommended as the source of values for the elements. Some time ago the Usage Board considered a proposal for additions to the DCMI Type vocabulary. At that time the Board decided that it did not see itself either entertaining additional proposals for expansion of DCMITYPE or managing additional vocabularies of values to be used with DC elements and element refinements. The Usage Board has two main concerns about expanding the number of vocabulary encoding schemes managed by DCMI:

* The resources required to manage vocabularies at this level is considerable. The Usage Board is composed of up to nine members all volunteers with busy careers on the side -- who feel a sufficient enough challenge

fulfilling the Board's primary mission;

* Most of the suggestions or proposals for additional vocabularies are those that should be created and managed by specialist communities with sufficient expertise and motivation to do the job.

To a great extent, the process of developing vocabularies is a social task, and not a technical task. As such it is critical that the development be managed in the appropriate domain. Usage Board members do not believe that the Usage Board is the appropriate domain for such an activity.

The Usage Board decided to reserve its decision on the proposed encoding schemes and to seek clarification of the role of the Board in this area, from the Trustees. Consequently, references to specific encoding schemes in the element proposals were removed.

Approved text - beginning

VMS-ID:	accrualMethod-001
Name:	accrualMethod
URI:	http://purl.org/dc/terms/accrualMethod
Namespace:	http://purl.org/dc/terms/
Label:	Accrual Method
Definition:	The method by which items are added to a collection.
Comment:	Recommended best practice is to use a value from an encoding scheme.
Type of term:	http://dublincore.org/usage/documents/principles/#element
Status:	http://dublincore.org/usage/documents/process/#conforming
Date issued:	2005-05-02
Decision:	http://dublincore.org/usage/decisions/#Decision-2005-02

Approved text - end

Approved text - beginning

VMS-ID:	accrualPeriodicity-001
Name:	accrualPeriodicity
URI:	http://purl.org/dc/terms/accrualPeriodicity
Namespace:	http://purl.org/dc/terms/
Label:	Accrual Periodicity
Definition:	The frequency with which items are added to a collection.
Comment:	Recommended best practice is to use a value from an encoding scheme.
Type of term:	http://dublincore.org/usage/documents/principles/#element
Status:	http://dublincore.org/usage/documents/process/#conforming
Date issued:	2005-05-02
Decision:	http://dublincore.org/usage/decisions/#Decision-2005-02

Approved text - end

Approved text - beginning

VMS-ID:	accrualPolicy-001
Name:	accrualPolicy
URI:	http://purl.org/dc/terms/accrualPolicy
Namespace:	http://purl.org/dc/terms/
Label:	Accrual Policy
Definition:	The policy governing the addition of items to a collection.
Comment:	Recommended best practice is to use a value from an encoding scheme.
Type of term:	http://dublincore.org/usage/documents/principles/#element
Status:	http://dublincore.org/usage/documents/process/#conforming
Date issued:	2005-05-02
Decision:	http://dublincore.org/usage/decisions/#Decision-2005-02

Approved text - end

Title:	**Collection Type (CLDType) Vocabulary**
Creator:	Dublin Core Collection Description Working Group
Date Issued:	2005-03-19
Identifier:	http://www.ukoln.ac.uk/metadata/dcmi/collection-type/2005-03-19/
Replaces:	http://www.ukoln.ac.uk/metadata/dcmi/collection-type/2003-11-10/
Is Replaced By:	Not applicable
Latest Version:	http://www.ukoln.ac.uk/metadata/dcmi/collection-type/
Status of Document:	This is a DCMI Working Draft.
Description of Document:	This document describes the collection type vocabulary used by the application profile for collection-level description developed by the Dublin Core Collection Description Working Group.

Introduction

The Collection Type (CLDType) Vocabulary provides a set of classes that can be used as values of the `dc:type` property in descriptions of collections. It has been developed in association with the Dublin Core Collection Description Application Profile (DC CD AP) but this vocabulary is independent of that profile and the classes defined here may be referenced in other metadata descriptions

The Collection Type (CLDType) Vocabulary allows for the categorisation of collections in two different ways:

- according to whether the collection is a catalogue or finding-aid for another collection
- according to the content of the items that make up the collection

All of these classes are defined as subclasses of the more general class `dcmitype:Collection`; i.e. a statement that a resource is an instance of **any** of the classes below implies also that the resource is an instance of the class `dcmitype:Collection`.

A metadata description may include multiple statements referencing the `dc:type` property, so a single collection may be an instance of several of these classes.

Note

Where the names are listed below as Qualified Names, prefixes are assumed to be associated with Namespace Names as follows:

- **dc:** `http:/purl.org/dc/elements/1.1/`
- **dcterms:** `http://purl.org/dc/terms/`

- **dcmitype:** http://purl.org/dc/dcmitype/
- **cld:** (collection-specific properties: URIs to be confirmed, temporary Namespace Name/URI http://example.org/cld/terms#)
- **cldtype:** (collection type terms: URIs to be confirmed, temporary Namespace Name/URI http://example.org/cld/type#)

Please note that where terms have Qualified Names with the prefixes cld and cldtype, this indicates that those terms have not yet been assigned persistent URIs. Until such persistent URIs are assigned, by DCMI or by some other naming authority, these terms should be considered to be unstable and should not be referenced in metadata descriptions, except as part of the evaluation/testing of this vocabulary.

Collection Type Vocabulary [cld:CLDType]

Identifier	**http://example.org/cld/terms#CLDType**
Name	CLDType
Label	Collection Type Vocabulary
Defined By	Collection Description Terms http://example.org/cld/terms#
Definition	A list of types that categorize a collection according to the nature of the content of the items in the collection and/or according to whether the collection is a catalogue, finding-aid or index for a second collection.
Comments	
See Also	DCMI Type Vocabulary http://dublincore.org/documents/dcmi-type-vocabulary/
See Also	An Analytical Model of Collections and their Catalogues http://www.ukoln.ac.uk/metadata/rslp/model/
Type of Term	Class/Vocabulary Encoding Scheme
Encoding Scheme for	Type, Dublin Core Metadata Element Set, v1.1 [dc:type] http://purl.org/dc/elements/1.1/type

Vocabulary Terms

By Type of Collection/Catalogue

Catalogue [cldtype:Catalogue]

Identifier	**http://example.org/cld/type#Catalogue**
Name	Catalogue
Label	Catalogue
Defined By	Collection Type Vocabulary http://example.org/cld/type#
Definition	A collection of individual records describing the items, and the intellectual content of those items, of a second collection. There may, in the individual records, be information about collections but that is not the focus of the catalogue. Catalogues are typically created with significant human input.
Comments	
See Also	An Analytical Model of Collections and their Catalogues http://www.ukoln.ac.uk/metadata/rslp/model/
Type of Term	Class/Vocabulary Term

SubClass Of	Collection, Dublin Core Type Vocabulary [dcmitype:Collection] http://purl.org/dc/dcmitype/Collection

Hierarchic Finding Aid [cldtype:HierarchicFindingAid]

Identifier	**http://example.org/cld/type#HierarchicFindingAid**
Name	HierarchicFindingAid
Label	Hierarchic Finding Aid
Defined By	Collection Type Vocabulary http://example.org/cld/type#
Definition	A collection of records describing the individual items, and the intellectual content of thos items, of a second collection. The records are firmly grounded within the overall arrangement of the collection, e.g. grouping together all the letters, account books etc. in an ordered sequence or sequences. Items are often not uniquely identifiable when considered in isolation, so the context of the collection is an essential element in compiling the hierarchic finding-aid. Hierarchic finding-aids are typically created with significant human input.
Comments	
See Also	An Analytical Model of Collections and their Catalogues http://www.ukoln.ac.uk/metadata/rslp/model/
Type of Term	Class/Vocabulary Term
SubClass Of	Collection, Dublin Core Type Vocabulary [dcmitype:Collection] http://purl.org/dc/dcmitype/Collection

Index [cldtype:Index]

Identifier	**http://example.org/cld/type#Index**
Name	Index
Label	Index
Defined By	Collection Type Vocabulary http://example.org/cld/type#
Definition	A collection of records consisting of information derived from items in a second collection, regardless of the content of those items. By this is meant that an index - for example a robotic search engine - will index the words in a document (or catalogue record) regardless of their context and without trying to identify the discrete elements of intellectual content contained therein. Indexes are typically generated automatically by a software robot or other harvesting technology.
Comments	
See Also	An Analytical Model of Collections and their Catalogues http://www.ukoln.ac.uk/metadata/rslp/model/
Type of Term	Class/Vocabulary Term
SubClass Of	Collection, Dublin Core Type Vocabulary [dcmitype:Collection] http://purl.org/dc/dcmitype/Collection

By Content of Items within Collection

Collection of Collections [cldtype:CollectionCollection]

Identifier	**http://example.org/cld/type#CollectionCollection**
Name	CollectionCollection
Label	Collection of Collections
Defined By	Collection Type Vocabulary http://example.org/cld/type#
Definition	A collection of items that are of type Collection, where a Collection is an aggregation of items; i.e. a collection of collections.
Comments	
See Also	
Type of Term	Class/Vocabulary Term
SubClass Of	Collection, Dublin Core Type Vocabulary [dcmitype:Collection] http://purl.org/dc/dcmitype/Collection

Collection of Datasets [cldtype:CollectionDataset]

Identifier	**http://example.org/cld/type#CollectionDataset**
Name	CollectionDataset
Label	Collection of Datasets
Defined By	Collection Type Vocabulary http://example.org/cld/type#
Definition	A collection of items of type Dataset where a Dataset is information encoded in a defined structure (for example, lists, tables, and databases), intended to be useful for direct machine processing.
Comments	
See Also	
Type of Term	Class/Vocabulary Term
SubClass Of	Collection, Dublin Core Type Vocabulary [dcmitype:Collection] http://purl.org/dc/dcmitype/Collection

Collection of Events [cldtype:CollectionEvent]

Identifier	**http://example.org/cld/type#CollectionEvent**
Name	CollectionEvent
Label	Collection of Events
Defined By	Collection Type Vocabulary http://example.org/cld/type#
Definition	A collection of items of type Event where an Event is is a non-persistent, time-based occurrence.
Comments	
See Also	
Type of Term	Class/Vocabulary Term
SubClass Of	Collection, Dublin Core Type Vocabulary [dcmitype:Collection] http://purl.org/dc/dcmitype/Collection

Collection of Images [cldtype:CollectionImage]

Identifier	**http://example.org/cld/type#CollectionImage**
Name	CollectionImage
Label	Collection of Images
Defined By	Collection Type Vocabulary http://example.org/cld/type#
Definition	A collection of items of type Image, where an Image is a primarily symbolic visual representation other than text. For example - images and photographs of physical objects, paintings, prints, drawings, other images and graphics, animations and moving pictures, film, diagrams, maps, musical notation. Note that image may include both electronic and physical representations.
Comments	
See Also	
Type of Term	Class/Vocabulary Term
SubClass Of	Collection, Dublin Core Type Vocabulary [dcmitype:Collection] http://purl.org/dc/dcmitype/Collection

Collection of Moving Images [cldtype:CollectionMovingImage]

Identifier	**http://example.org/cld/type#CollectionMovingImage**
Name	CollectionMovingImage
Label	Collection of Moving Images
Defined By	Collection Type Vocabulary http://example.org/cld/type#
Definition	A collection of items of type Moving Image, where a Moving Image is a series of visual representations that, when shown in succession, impart an impression of motion. For example, an animation, movie, television, video, zoetrope, or visual output from a simulation.
Comments	
See Also	
Type of Term	Class/Vocabulary Term
SubClass Of	CollectionImage, Collection Type Vocabulary [cldtype:CollectionImage] http://example.org/cldtype/CollectionImage

Collection of Still Images [cldtype:CollectionStillImage]

Identifier	**http://example.org/cld/type#CollectionStillImage**
Name	CollectionStillImage
Label	Collection of Still Images
Defined By	Collection Type Vocabulary http://example.org/cld/type#
Definition	A collection of items of type Still Image, where a Still Image is a a static visual representation other than text. For example, a picture, photograph, painting, drawing, graphic design, plan, map, or musical score. Note that a Still Image may be manifested in physical and electronic forms.
Comments	
See Also	
Type of Term	Class/Vocabulary Term

SubClass Of	CollectionImage, Collection Type Vocabulary [cldtype:CollectionImage] http://example.org/cldtype/CollectionImage

Collection of Interactive Resources [cldtype:CollectionInteractiveResource]

Identifier	**http://example.org/cld/type#CollectionInteractiveResource**
Name	CollectionInteractiveResource
Label	Collection of Interactive Resources
Defined By	Collection Type Vocabulary http://example.org/cld/type#
Definition	A collection of items of type Interactive Resource, where an an interactive resource is a resource which requires interaction from the user to be understood, executed, or experienced. For example, forms on web pages, applets, multimedia learning objects, chat services, virtual reality.
Comments	
See Also	
Type of Term	Class/Vocabulary Term
SubClass Of	Collection, Dublin Core Type Vocabulary [dcmitype:Collection] http://purl.org/dc/dcmitype/Collection

Collection of Physical Objects [cldtype:CollectionPhysicalObject]

Identifier	**http://example.org/cld/type#CollectionPhysicalObject**
Name	CollectionPhysicalObject
Label	Collection of Physical Objects
Defined By	Collection Type Vocabulary http://example.org/cld/type#
Definition	A collection of items of type Physical Object, where a Physical Object is an inanimate, three-dimensional object or substance. For example, a computer, the great pyramid, a sculpture.
Comments	
See Also	
Type of Term	Class/Vocabulary Term
SubClass Of	Collection, Dublin Core Type Vocabulary [dcmitype:Collection] http://purl.org/dc/dcmitype/Collection

Collection of Services [cldtype:CollectionService]

Identifier	**http://example.org/cld/type#CollectionService**
Name	CollectionService
Label	Collection of Services
Defined By	Collection Type Vocabulary http://example.org/cld/type#
Definition	A collection of items of type Service, where a Service is a system that provides one or more functions of value to the end-user. Examples of Services include: a photocopying service, a banking service, an authentication service, interlibrary loans, a Z39.50 or Web server.

Comments	
See Also	
Type of Term	Class/Vocabulary Term
SubClass Of	Collection, Dublin Core Type Vocabulary [dcmitype:Collection] http://purl.org/dc/dcmitype/Collection

Collection of Software [cldtype:CollectionSoftware]

Identifier	**http://example.org/cld/type#CollectionSoftware**
Name	CollectionSoftware
Label	Collection of Software
Defined By	Collection Type Vocabulary http://example.org/cld/type#
Definition	A collection of items of type Software, where Software is a computer program in source or compiled form which may be available for installation non-transiently on another machine.
Comments	
See Also	
Type of Term	Class/Vocabulary Term
SubClass Of	Collection, Dublin Core Type Vocabulary [dcmitype:Collection] http://purl.org/dc/dcmitype/Collection

Collection of Sounds [cldtype:CollectionSound]

Identifier	**http://example.org/cld/type#CollectionSound**
Name	CollectionSound
Label	Collection of Sounds
Defined By	Collection Type Vocabulary http://example.org/cld/type#
Definition	A collection of items of type Sound, where a Sound is a resource whose content is primarily intended to be rendered as audio. For example - a music playback file format, an audio compact disc, and recorded speech or sounds.
Comments	
See Also	
Type of Term	Class/Vocabulary Term
SubClass Of	Collection, Dublin Core Type Vocabulary [dcmitype:Collection] http://purl.org/dc/dcmitype/Collection

Collection of Texts [cldtype:CollectionText]

Identifier	**http://example.org/cld/type#CollectionText**
Name	CollectionText
Label	Collection of Texts
Defined By	Collection Type Vocabulary http://example.org/cld/type#

Definition	A collection of items of type Text, where a Text is a resource whose content is primarily words for reading. For example - books, letters, dissertations, poems, newspapers, articles, archives of mailing lists. Note that facsimiles or images of texts are still of the genre text.
Comments	
See Also	
Type of Term	Class/Vocabulary Term
SubClass Of	Collection, Dublin Core Type Vocabulary [dcmitype:Collection] http://purl.org/dc/dcmitype/Collection

Changes made in this version

- Add introductory text.
- Add "Catalogue", "Finding-Aid" and "Index".
- Alter labels for collection types based on item types to "Collection of....".
- Add "Collection of Collections", "Collection of Events" and "Collection of Services".

Metadata associated with this resource: http://dublincore.org/documents/collection-type/index.shtml.rdf

Title:	**Dublin Core Collection Description Proposed Term : DCCDAccrualMethod**
Creator:	Dublin Core Collection Description Working Group
Date Issued:	2004-07-30
Identifier:	http://www.ukoln.ac.uk/metadata/dcmi/collection-DCCDAccrualMethod/2004-07-30/
Replaces:	Not applicable
Is Replaced By:	Not applicable
Latest Version:	http://www.ukoln.ac.uk/metadata/dcmi/collection-DCCDAccrualMethod/
Status of Document:	This is a DCMI Working Draft.
Description of Document:	This document presents a proposal from the Dublin Core Collection Description Working Group for a new encoding scheme, DCCDAccrualMethod.

Proposal

Name	**DCCDAccrualMethod**	
Label	DCCD Accrual Method	
Definition	Methods by which items are added to a collection.	
Values	Deposit	The permanent addition of items to the collection, where the transfer of ownership is conditional on certain requirements or restrictions, but without financial payment or reciprocal transfer of items.
	Donation	The permanent addition of items to the collection through the transfer of ownership, without financial payment.
	Purchase	The permanent addition of items to the collection through the transfer of ownership, accompanied by one or more financial payments.
	Loan	The temporary addition of items to the collection with no transfer of ownership, without financial payment.
	License	The temporary addition of items to the collection with no transfer of ownership, accompanied by one or more financial payments.

	ItemCreation The permanent addition of items to the collection as a result of item creation by the owner of the collection.
Type of term	Vocabulary Encoding Scheme
Term qualified	(Proposed element) accrualMethod See http://www.ukoln.ac.uk/metadata/dcmi/collection-accrualMethod/
Why needed	This encoding scheme allows the unambiguous specification of values for the proposed element accrualMethod, which in turn supports precise searching.
Working Group support	See the mailing list archives of the WG, especially Jul 2004.
Proposed status	Registered
Related DCMI terms	[n/a]
Related non-DCMI terms	(Proposed element) accrualMethod See http://www.ukoln.ac.uk/metadata/dcmi/collection-accrualMethod/
About the proposers	The term is proposed by the Dublin Core Collection Description Working Group. One of the primary aims of the WG is the development of a Dublin Core Application Profile (DCAP) for collection-level description, i.e. for the description of a collection as a resource, rather than the description of the individual items that make up that collection. Records of the activity of the WG are available in the mailing list archives. The current draft of the Collection Description Application Profile is available at http://www.ukoln.ac.uk/metadata/dcmi/collection-application-profile/

Changes made in this version

- Initial version.

Metadata associated with this resource: http://dublincore.org/documents/collection-DCCDAccrualMethod/index.shtml.rdf

Title:	**Dublin Core Collection Description Proposed Term : DCCDAccrualPeriodicity**
Creator:	Dublin Core Collection Description Working Group
Date Issued:	2004-08-18
Identifier:	http://www.ukoln.ac.uk/metadata/dcmi/collection-DCCDAccrualPeriodicity/2004-08-18/
Replaces:	http://www.ukoln.ac.uk/metadata/dcmi/collection-DCCDAccrualPeriodicity/2004-07-30/
Is Replaced By:	Not applicable
Latest Version:	http://www.ukoln.ac.uk/metadata/dcmi/collection-DCCDAccrualPeriodicity/
Status of Document:	This is a DCMI Working Draft.
Description of Document:	This document presents a proposal from the Dublin Core Collection Description Working Group for a new encoding scheme, DCCDAccrualPeriodicity.

Proposal

Name	**DCCDAccrualPeriodicity**
Label	DCCD Accrual Periodicity
Definition	Frequencies with which items are added to a collection.
Values	Annual Bimonthly Semiweekly Daily Biweekly Semiannual

	Biennial Triennial Three times a week Three times a month Continuously updated Monthly Quarterly Semimonthly Three times a year Weekly Completely irregular
Type of term	Vocabulary Encoding Scheme
Term qualified	(Proposed element) accrualPeriodicity See http://www.ukoln.ac.uk/metadata/dcmi/collection-accrualPeriodicity/
Why needed	This encoding scheme allows the unambiguous specification of values for the proposed element accrualPeriodicity, which in turn supports precise searching.
Working Group support	See the mailing list archives of the WG, especially Jul 2004.
Proposed status	Registered
Related DCMI terms	This is the same set of values as used in MARC21. See *MARC 21 Concise Holdings: Caption and Pattern Fields (853-855)* (subfield $w Frequency - Codes used for frequencies that have a fundamental periodicity). However, there is no evidence that MARC has issued URIrefs to refer either to the scheme as a unit or to the individual values within the scheme.

Related non-DCMI terms	(Proposed element) accrualPeriodicity See http://www.ukoln.ac.uk/metadata/dcmi/collection-accrualPeriodicity/
About the proposers	The term is proposed by the Dublin Core Collection Description Working Group. One of the primary aims of the WG is the development of a Dublin Core Application Profile (DCAP) for collection-level description, i.e. for the description of a collection as a resource, rather than the description of the individual items that make up that collection. Records of the activity of the WG are available in the mailing list archives. The current draft of the Collection Description Application Profile is available at http://www.ukoln.ac.uk/metadata/dcmi/collection-application-profile/

Changes made in this version

- Added reference to MARC 21 frequency codes.

Metadata associated with this resource: http://dublincore.org/documents/collection-DCCDAccrualPeriodicity/index.shtml.rdf

Title:	**Dublin Core Collection Description Proposed Term : DCCDAccrualPolicy**
Creator:	Dublin Core Collection Description Working Group
Date Issued:	2004-07-30
Identifier:	http://www.ukoln.ac.uk/metadata/dcmi/collection-DCCDAccrualPolicy/2004-07-30/
Replaces:	Not applicable
Is Replaced By:	Not applicable
Latest Version:	http://www.ukoln.ac.uk/metadata/dcmi/collection-DCCDAccrualPolicy/
Status of Document:	This is a DCMI Working Draft.
Description of Document:	This document presents a proposal from the Dublin Core Collection Description Working Group for a new encoding scheme, DCCDAccrualPolicy.

Proposal

Name	**DCCDAccrualPolicy**
Label	DCCD Accrual Policy
Definition	Policies governing the addition of items to a collection.
Values	Closed A policy that items are no longer added to the collection. Passive A policy that items are added to the collection only in response to the initiative of an external agent. Active A policy that items are actively sought for addition to the collection. Partial A policy that items are actively sought for addition to a specific part of the collection.
Type of term	Vocabulary Encoding Scheme
Term qualified	(Proposed element) accrualPolicy See http://www.ukoln.ac.uk/metadata/dcmi/collection-accrualPolicy/

Why needed	This encoding scheme allows the unambiguous specification of values for the proposed element `accrualPolicy`, which in turn supports precise searching.
Working Group support	See the mailing list archives of the WG, especially Jul 2004.
Proposed status	Registered
Related DCMI terms	[n/a]
Related non-DCMI terms	(Proposed element) `accrualPolicy` See `http://www.ukoln.ac.uk/metadata/dcmi/collection-accrualPolicy/`
About the proposers	The term is proposed by the Dublin Core Collection Description Working Group. One of the primary aims of the WG is the development of a Dublin Core Application Profile (DCAP) for collection-level description, i.e. for the description of a collection as a resource, rather than the description of the individual items that make up that collection. Records of the activity of the WG are available in the mailing list archives. The current draft of the Collection Description Application Profile is available at `http://www.ukoln.ac.uk/metadata/dcmi/collection-application-profile/`

Changes made in this version

- Initial version.

Metadata associated with this resource: http://dublincore.org/documents/collection-DCCDAccrualPolicy/index.shtml.rdf

Library Application Profile

Creator: DCMI-Libraries Working Group
Contributors: Robina Clayphan, The British Library, UK
Rebecca Guenther, Library of Congress, USA
Date Issued: 2004-09-10
Identifier: http://dublincore.org/documents/2004/09/10/library-application-profile/
Replaces: http://dublincore.org/documents/2002/09/24/library-application-profile/
Is Replaced By: Not applicable
Latest Version: http://dublincore.org/documents/2004/09/10/library-application-profile/
Status of Document: This is a DCMI Working Draft.
Description of Document: This document proposes a possible application profile that clarifies the use of the Dublin Core Metadata Element Set in libraries and library-related applications and projects. It was originally prepared by the DCMI-Libraries Application Profile drafting committee, a subset of the DCMI-Libraries Working Group.
This revision was prepared in August 2004 and incorporates decisions made by the DCMI Usage Board at its meetings in 2003 and issues discussed in the WG meeting in Seattle in September 2003. It has been reformatted in conformance with the Dublin Core Application Profile Guidelines produced by the CEN MMI-DC Workshop.

DC-Library Application Profile (DC-Lib)

I. Introduction

The concept of *application profiles* (see Application profiles: mixing and matching metadata schemas) has emerged within the Dublin Core Metadata Initiative as a way to declare which elements from which namespaces are used in a particular application or project. Application profiles are defined as schemas which consist of data elements drawn from one or more namespaces, combined together by implementors, and optimised for a particular local application.

The DCMI-Libraries Working Group has explored various uses of the Dublin Core Metadata Element Set in library and related applications and has envisioned the following possible uses:

- to serve as an interchange format between various systems using different metadata standards/formats
- to use for harvesting metadata from data sources within and outside of the library domain
- to support simple creation of library catalog records for resources within a variety of systems
- to expose MARC data to other communities (through a conversion to DC)
- to allow for acquiring resource discovery metadata from non-library creators using DC

A *library application profile* will be a specification that defines the following:

- required elements
- permitted Dublin Core elements
- permitted Dublin Core qualifiers
- permitted schemes and values (e.g. use of a specific controlled vocabulary or encoding scheme)
- library domain elements used from another namespace
- additional elements/qualifiers from other application profiles that may be used (e.g. DC-Education: Audience)
- refinement of standard definitions

This document proposes a possible application profile that clarifies the use of the Dublin Core Metadata Element Set in libraries and library-related applications and projects. It was originally prepared by the DCMI-Libraries Application Profile drafting committee, a subset of the DCMI-Libraries WorkingGroup. This revision was prepared in August 2004 and incorporates decisions made by the DCMI Usage Board at its meetings in 2003 and issues discussed in the WG meeting in Seattle in September 2003. It has been reformatted in conformance with the Dublin Core Application Profile Guidelines produced by the CEN MMI-DC Workshop.

2. Namespaces and Format of entries

The DC-Library Application Profile uses terms from two namespaces:

- DCMI Metadata Terms [http://dublincore.org/documents/dcmi-terms/]
- MODS elements used in DC-Lib application profile [http://www.loc.gov/mods]
- The Usage Board has decided that any encoding scheme that has a URI defined in a non-DCMI namespace may be used. DCMI will not define these. Those so far considered for this application profile are indicated.

Format of entries:

This Application Profile is presented following the "Dublin Core Application Profiles Guidelines" produced by the CEN MMI-DC Workshop.

Name of Term	A unique token assigned to the term
Term URI	A Uniform Resource Identifier used to identify the term.
Label	A human-readable label assigned to the term.
Defined By	An identifier of a namespace, pointer to a schema, or bibliographic reference for a document within which the term is defined.
Source Definition	The definition of the term in the namespace in which the term was orginated.
DC-Lib Definition	The DC-Lib definition of the term.
Source Comments	Comments on the term from the namespace in which the term orginated.
DC-Lib Comments	DC-Lib comments about the term.
Type of term	The grammatical category of the term (e.g. "Element", "Element Refinement", or "Encoding Scheme").
Refines	The described term semantically refines the referenced term. A refinement makes the meaning of the element narrower or more specific. It will share the meaning of the unrefined element but with a more restricted scope.
Refined By	The described term is semantically refined by the referenced term.
Encoding Scheme For	The described term, an encoding scheme, qualifies the referenced term. Using an encoding scheme will aid in the interpretation of an element value. These schemes include controlled vocabularies and formal notations or parsing rules. A value expressed using an encoding scheme will thus be a token selected from a controlled vocabulary (e.g., a term from a classification system or set of subject headings) or a string formatted in accordance with a formal notation (e.g., "2000-01-01" as the standard expression of a date). If an encoding scheme is not understood by a client or agent, the value may still be useful to a human reader. In some cases, encoding schemes not yet registered are indicated. These will be registered and/or approved by the DCMI Usage Board as DC Encoding Schemes in the future.
Has Encoding Scheme	The described term is qualified by the referenced encoding scheme.
Obligation	Indicates whether the element is required to always or sometimes be present. In this application profile the obligation can be: mandatory (**M**), mandatory if applicable (**MA**), strongly recommended (**R**) or optional (**O**). Mandatory ensures that some of the elements are always supported and mandatory if applicable means that this element must be supported if the information is available. An element with a mandatory obligation must have a value. The strongly recommended and the optional elements should be filled with a value if the information is appropriate to the given resource but if not, they may be omitted.
Occurence	Indicates any limit to the repeatability of the element.

3. Table of Contents

- Relation
- IsVersionOf
- IsFormatOf
- HasFormat
- IsReplacedBy
- Replaces
- IsPartOf
- HasPart
- Requires
- IsReferencedBy
- References
- Coverage
- Spatial
- Temporal
- Rights
- Audience
- Edition
- Location

General notes regarding all elements:

- Either a Title or Identifier are mandatory (Identifier is mandatory if applicable).
- If the record is expressed in a format (e.g., HTML) that allows each element/qualifier/scheme value to be assigned a language attribute (e.g., the HTML tag lang="en"), the use of the attribute is permitted for any or all DC elements as desired.
- All elements may be used as unqualified. If using qualified Dublin Core, additional guidelines are given.
- Any encoding scheme that has a URI defined in a non-DCMI namespace may be used: DCMI will not define these. A mechanism to register these is awaited. Where applicable a table has been created for each encoding scheme in a separate section following the main body of tables for elements and refinements.

4. DC-Library Application Profile

Name of Term	**title**
Term URI	http://purl.org/dc/elements/1.1/title
Label	Title
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	A name given to the resource
DC-Lib Definition	-
Source Comments	Typically, a title will be a name by which the resource is formally known.
DC-Lib Comments	A parallel/transliterated title is considered a main title, i.e. the Title element is repeated. Either a title or identifier is mandatory. If no title is available, best practice is to give a constructed title, derive a title from the resource or supply *[no title]*. If using qualified Dublin Core, an element refinement for titles other than the main title(s) should be included. Retain initial articles and use local sorting algorithms based on language. A language qualifier may be used to indicate language of title if appropriate. (For example, see: Initial Definite and Indefinite Articles for a list of articles in various languages).
Type of term	element
Refines	
Refined By	alternative
Has Encoding Scheme	
Obligation	**M**
Occurence	

Name of Term	**alternative**
Term URI	http://purl.org/dc/terms/alternative
Label	Alternative
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	Any form of the title used as a substitute or alternative to the formal title of the resource.
DC-Lib Definition	-
Source Comments	This qualifier can include Title abbreviations as well as translations.

DC-Lib Comments	Assigned title such as uniform or key title is Alternative. Best practice is to use this element refinement for titles other than the main title. Retain initial articles and use local sorting algorithms based on language. A language qualifier may be used to indicate language of title if appropriate. (For example, see: Initial Definite and Indefinite Articles for a list of articles in various languages)
Type of term	element refinement
Refines	title
Refined By	
Has Encoding Scheme	
Obligation	**R**
Occurence	

Back to TOC

Name of Term	**creator**
Term URI	http://purl.org/dc/elements/1.1/creator
Label	Creator
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	An entity primarily responsible for making the content of the resource.
DC-Lib Definition	An entity with a primary role in the creation of the intellectual or artistic content of the resource.
Source Comments	Examples of a Creator include a person, an organisation, or a service. Typically, the name of a Creator should be used to indicate the entity.
DC-Lib Comments	This term should not be refined with a role. A subset of terms taken from the Library of Congress list of Relators have been approved as role refinements for Contributor (http://www.loc.gov/marc/sourcecode/relator/relatorlist.html). URIs will be provided when available. The DCMI Usage Board has disapproved the idea of structured values to provide more information about the creator - it should be in a description for another resource. Creator and Contributor may be conflated with Creator being used as a refinement of Contributor.
Type of term	element
Refines	
Refined By	
Has Encoding Scheme	
Obligation	**O**
Occurence	

Name of Term	**contributor**
Term URI	http://purl.org/dc/elements/1.1/contributor
Label	Contributor
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	An entity responsible for making contributions to the content of the resource.
DC-Lib Definition	-
Source Comments	Examples of a Contributor include a person, an organisation, or a service. Typically, the name of a Contributor should be used to indicate the entity.
DC-Lib Comments	A subset of terms taken from the Library of Congress list of Relators have been approved as role refinements for Contributor (http://www.loc.gov/marc/sourcecode/relator/relatorlist.html). URIs will be provided when available. The DCMI Usage Board has disapproved the idea of structured values to provide more information about a contributor - it should be in a description for another resource. Creator and Contributor may be conflated with Creator being used as a refinement of Contributor.
Type of term	element
Refines	
Refined By	
Has Encoding Scheme	Role list (http://www.loc.gov/...)
Obligation	**MA**
Occurence	

Back to TOC

Name of Term	**publisher**
Term URI	http://purl.org/dc/elements/1.1/publisher
Label	Publisher
Defined By	http://dublincore.org/documents/dcmi-terms/

Source Definition	An entity responsible for making the resource available.
DC-Lib Definition	
Source Comments	Examples of a Publisher include a person, an organisation, or a service. Typically, the name of a Publisher should be used to indicate the entity.
DC-Lib Comments	A subset of terms taken from the Library of Congress list of Relators has been approved for use as role refinements of Publisher if applicable (http://www.loc.gov/marc/sourcecode/relator/relatorlist.html). URIs will be provided when available. The DCMI Usage Board has disapproved the idea of structured values to provide more information about the publisher - it should be in a description for another resource. Although some applications may wish to conflate Creator, Contributor and Publisher, DC-Lib maintains the distinction between Creator and Contributor (which may be conflated) and Publisher. If the elements are conflated and Publisher used as an element refinement for Contributor, the resulting element would be mapped to DC.Publisher.
Type of term	element
Refines	
Refined By	
Has Encoding Scheme	Role list (http://www.loc.gov/...)
Obligation	**O**
Occurence	

Name of Term	**subject**
Term URI	http://purl.org/dc/elements/1.1/subject
Label	Subject
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	The topic of the content of the resource.
DC-Lib Definition	
Source Comments	Typically, a Subject will be expressed as keywords, key phrases or classification codes that describe a topic of the resource. Recommended best practice is to select a value from a controlled vocabulary or formal classification scheme.
DC-Lib Comments	If a geographic or temporal aspect is recorded use the element Coverage. It may also be repeated in Subject if desired. If there is a subject string with elements in addition to a geographic, include the entire string in Subject with geographic element also in Coverage. It is highly recommended that either freetext or controlled vocabulary be supplied as element Subject in the metadata where appropriate and feasible. It is also recommended that a controlled vocabulary be used with encoding scheme specified. If no encoding scheme is specified, it is treated as keyword. If using qualified DC, always use the encoding scheme(s) for terms from a controlled vocabulary.
Type of term	element
Refines	
Refined By	
Has Encoding Scheme	Library of Congress Subject Headings - http://purl.org/dc/terms/LCSH Medical Subject Headings - http://purl.org/dc/terms/MESH Dewey Decimal Classification - http://purl.org/dc/terms/DDC Library of Congress Classification - http://purl.org/dc/terms/LCC Universal Dewey Classification - http://purl.org/dc/terms/UDC These are encoding schemes currently defined by DCMI. As additional schemes are registered, they will be included. Additional encoding schemes will be registered for those used in the library domain based on the MARC list of subject and classification schemes. Including an identifier to link to a registry where all encoding schemes are defined (e.g. based on RSLP schema) needs to be explored.
Obligation	**MA**
Occurence	

Back to TOC

Name of Term	**description**
Term URI	http://purl.org/dc/elements/1.1/description

Label	Description
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	An account of the content of the resource.
DC-Lib Definition	
Source Comments	Description may include but is not limited to: an abstract, table of contents, reference to a graphical representation of content or a free-text account of the content.
DC-Lib Comments	It is permitted to link one or more external descriptions using a URI, but to facilitate keyword indexing of the content of the description, it is recommended that a text description also be included.
Type of term	element
Refines	
Refined By	Abstract, tableOfContents
Has Encoding Scheme	URI - http://purl.org/dc/terms/URI
Obligation	**R**
Occurence	

Name of Term	**abstract**
Term URI	http://purl.org/dc/terms/abstract
Label	Abstract
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	An account of the content of the resource.
DC-Lib Definition	
Source Comments	
DC-Lib Comments	Use text (and not only a URL) to describe the resource.
Type of term	element refinement
Refines	Description
Refined By	
Has Encoding Scheme	URI - http://purl.org/dc/terms/URI
Obligation	**R**
Occurence	

Name of Term	**tableOfContents**
Term URI	http://purl.org/dc/terms/tableOfContents
Label	Table Of Contents
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	A list of subunits of the content of the resource
DC-Lib Definition	
Source Comments	
DC-Lib Comments	Use text (and not only a URL) to describe the resource.
Type of term	element refinement
Refines	Description
Refined By	
Has Encoding Scheme	URI - http://purl.org/dc/terms/URI
Obligation	**R**
Occurence	

Back to TOC

Name of Term	**date**
Term URI	http://purl.org/dc/elements/1.1/date
Label	Date
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	A date associated with an event in the life cycle of the resource.
DC-Lib Definition	
Source Comments	Typically, date will be associated with the creation or availability of the resource. Recommended best practice for encoding the date value is defined in a profile of ISO 8601 [W3CDTF] and follows the YYYY-MM-DD format.
DC-Lib Comments	Recommend use of an element refinement for type of Date. Recommend that dates be encoded: 1) using W3C-DTF (a profile of ISO 8601 structured with hyphens), 2) using ISO 8601 (structured without hyphens), or 3) supplied as free text that does not take the form of a string of numerals (with or without hyphens). The second option, ISO 8601 (without hyphens), is preferred. It is acceptable to use widely-recognised practice such as day-month-year where the day and year are represented with numerals and month with a name or standard abbreviation (e.g., "1 January 2002" or "1 Jan 2002"). Avoid the use of potentially ambiguous date representations such as DD/MM/YY or MM/DD/YY (e.g., "04/05/05") It may be desirable to establish a DC-Lib encoding scheme or profile of ISO 8601 to cover B.C.E. dates, questionable and approximate dates. A date working group has been established to progress these issues.

Type of term	element
Refines	
Refined By	Created, Valid, Available, Issued, Modified
Has Encoding Scheme	ISO 8601(without hyphens) - http://purl.org/dc/terms/ISO8601 W3C-DTF (with hyphens) - http://purl.org/dc/terms/W3CDTF
Obligation	**MA**
Occurence	

Name of Term	**created**
Term URI	http://purl.org/dc/terms/created
Label	Created
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	Date of creation of the resource.
DC-Lib Definition	
Source Comments	
DC-Lib Comments	Use for the creation of the intellectual content.
Type of term	element refinement
Refines	Date
Refined By	
Has Encoding Scheme	ISO 8601(without hyphens) - http://purl.org/dc/terms/ISO8601 W3C-DTF (with hyphens) - http://purl.org/dc/terms/W3CDTF
Obligation	**MA**
Occurence	This qualified element should not be repeated except when giving date created using more than one encoding scheme.

Name of Term	**valid**
Term URI	http://purl.org/dc/terms/valid
Label	Valid
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	Date (often a range) of validity of the resource.
DC-Lib Definition	
Source Comments	
DC-Lib Comments	
Type of term	element refinement
Refines	Date
Refined By	
Has Encoding Scheme	ISO 8601(without hyphens) - http://purl.org/dc/terms/ISO8601 W3C-DTF (with hyphens) - http://purl.org/dc/terms/W3CDTF
Obligation	**O**
Occurence	

Name of Term	**available**
Term URI	http://purl.org/dc/terms/available
Label	Available
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	Date (often a range) that the resource will become or did become available.
DC-Lib Definition	
Source Comments	
DC-Lib Comments	
Type of term	element refinement
Refines	Date
Refined By	
Has Encoding Scheme	ISO 8601(without hyphens) - http://purl.org/dc/terms/ISO8601 W3C-DTF (with hyphens) - http://purl.org/dc/terms/W3CDTF
Obligation	**O**
Occurence	

Name of Term	**issued**
Term URI	http://purl.org/dc/terms/issued

Label	Issued
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	Date of formal issurance (e.g. publication) of the resource.
DC-Lib Definition	
Source Comments	
DC-Lib Comments	Use for the instantiation.
Type of term	element refinement
Refines	Date
Refined By	
Has Encoding Scheme	ISO 8601(without hyphens) - http://purl.org/dc/terms/ISO8601 W3C-DTF (with hyphens) - http://purl.org/dc/terms/W3CDTF
Obligation	**MA**
Occurence	

Name of Term	**modified**
Term URI	http://purl.org/dc/terms/modified
Label	Modified
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	Date on which resource was changed.
DC-Lib Definition	
Source Comments	
DC-Lib Comments	
Type of term	element refinement
Refines	Date
Refined By	
Has Encoding Scheme	ISO 8601(without hyphens) - http://purl.org/dc/terms/ISO8601 W3C-DTF (with hyphens) - http://purl.org/dc/terms/W3CDTF
Obligation	**O**
Occurence	

Name of Term	**dateCopyrighted**
Term URI	http://purl.org/dc/terms/dateCopyrighted
Label	Date Copyrighted
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	Date of copyright statement.
DC-Lib Definition	
Source Comments	
DC-Lib Comments	Decision by DCMI Usage Board May 2002. Recommend use if: 1) the value is different from Issued or Created, or 2) the copyright date is known but no value is supplied for Issued or Created. If same date is used for issued and copyright date, use only Issued.
Type of term	element refinement
Refines	Date
Refined By	
Has Encoding Scheme	ISO 8601(without hyphens) - http://purl.org/dc/terms/ISO8601 W3C-DTF (with hyphens) - http://purl.org/dc/terms/W3CDTF
Obligation	**O**
Occurence	

Name of Term	**dateSubmitted**
Term URI	http://purl.org/dc/terms/dateSubmitted
Label	Date Submitted
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	Date of submission of the resource (e.g. thesis, articles, etc.).
DC-Lib Definition	
Source Comments	
DC-Lib Comments	Decision by DCMI Usage Board in May 2002. Recommended for theses and dissertations.
Type of term	element refinement
Refines	Date
Refined By	
Has Encoding Scheme	ISO 8601(without hyphens) - http://purl.org/dc/terms/ISO8601 W3C-DTF (with hyphens) - http://purl.org/dc/terms/W3CDTF

Obligation	**O**
Occurence	

Name of Term	**dateAccepted**
Term URI	http://purl.org/dc/terms/dateAccepted
Label	Date Accepted
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	Date of acceptance of the resource (e.g. of thesis by university department/institution, of article by journal, etc.).
DC-Lib Definition	
Source Comments	
DC-Lib Comments	Decision by DCMI Usage Board in May 2002. Recommended for theses and dissertations.
Type of term	element refinement
Refines	Date
Refined By	
Has Encoding Scheme	ISO 8601(without hyphens) - http://purl.org/dc/terms/ISO8601 W3C-DTF (with hyphens) - http://purl.org/dc/terms/W3CDTF
Obligation	**O**
Occurence	

Name of Term	**dateCaptured**
Term URI	http://www.loc.gov/mods/
Label	Date Captured
Defined By	
Source Definition	
DC-Lib Definition	Date that the resource was captured.
Source Comments	
DC-Lib Comments	This includes the date that a snapshot of the resource was taken (particularly for dynamic resources) if different from Date.Created. Use existing element <dateCaptured> under <publicationInfo> in the Metadata Object Description Schema. Decision by DCMI Usage Board in May 2002. Best practice is to use as a machine-processible date (ISO 8601 without hyphens or W3CDTF with hyphens).
Type of term	element refinement
Refines	Date
Refined By	
Has Encoding Scheme	ISO 8601(without hyphens) - http://purl.org/dc/terms/ISO8601 W3C-DTF (with hyphens) - http://purl.org/dc/terms/W3CDTF
Obligation	**O**
Occurence	

Back to TOC

Name of Term	**type**
Term URI	http://purl.org/dc/elements/1.1/type
Label	Resource Type
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	The nature or genre of the content of the resource.
DC-Lib Definition	
Source Comments	Type includes terms describing general categories, functions, genres, or aggregation levels for content. Recommended best practice is to select a value from a controlled vocabulary (for example, the list of DCMI Types). To describe the physical or digital manifestation of the resource, use the Format element.
DC-Lib Comments	Use a controlled list and identify the source with encoding scheme. Recommended that at least one value from DCMI-Type be supplied for a high level category. Use a controlled list and identify the source with encoding scheme. Consider registering values defined in the MARC list of sources as encoding schemes as well as any others that are identified as useful.

Type of term	element
Refines	
Refined By	
Has Encoding Scheme	DCMIType - http://purl.org/dc/dcmitype/ The DCMI Type Vocabulary is a list of types used to categorize the nature or genre of the content of the resource. See also http://dublincore.org/documents/dcmi-type-vocabulary/ Values from MARC list of sources to be registered as encoding schemes Art & Architecture thesaurus - http://www.loc.gov/marc/source/aat Thesaurus for graphic materials - http://www.loc.gov/marc/source/gmgpc (Note: These URIs established by LOC are subject to confirmation).
Obligation	**O**
Occurence	Type may be repeated for each encoding scheme used.

Back to TOC

Name of Term	**format**
Term URI	http://purl.org/dc/elements/1.1/format
Label	Format
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	The physical or digital manifestation of the resource.
DC-Lib Definition	
Source Comments	Typically, Format may include the media-type or dimensions of the resource. Format may be used to determine the software, hardware or other equipment needed to display or operate the resource. Examples of dimensions include size and duration. Recommended best practice is to select a value from a controlled vocabulary (for example, the list of Internet Media Types [MIME] defining computer media formats).
DC-Lib Comments	Use this element primarily for IMT. Recommended for electronic resources.
Type of term	element
Refines	
Refined By	Extent, Medium
Has Encoding Scheme	IMT - http://purl.org/dc/terms/IMT The Internet media type of the resource. See also: http://www.isi.edu/in-notes/iana/assignments/media-types/media-types
Obligation	**R**
Occurence	

Name of Term	**extent**
Term URI	http://purl.org/dc/terms/extent
Label	Extent
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	The size or duration of the resource.
DC-Lib Definition	
Source Comments	
DC-Lib Comments	
Type of term	element refinement
Refines	Format
Refined By	
Has Encoding Scheme	
Similar to	
Obligation	**O**
Datatype	
Occurence	

Name of Term	**medium**
Term URI	http://purl.org/dc/terms/medium
Label	Medium
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	The material or physical carrier of the resource.
DC-Lib Definition	
Source Comments	

DC-Lib Comments	Used to specify the medium of the physical carrier of a resource. Format without an element refinement qualifier should be used to specify the electronic format of the resource, using the encoding scheme IMT. Format should be repeated if both are applicable (e.g. a PDF file on CD).
Type of term	element refinement
Refines	Format
Refined By	
Has Encoding Scheme	IMT - http://purl.org/dc/terms/IMT The Internet media type of the resource. See also: http://www.isi.edu/in-notes/iana/assignments/media-types/media-types
Obligation	**O**
Occurence	

Back to TOC

Name of Term	**identifier**
Term URI	http://purl.org/dc/elements/1.1/identifier
Label	Identifier
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	An unambiguous reference to the resource within a given context.
DC-Lib Definition	
Source Comments	Recommended best practice is to identify the resource by means of a string or number conforming to a formal identification system. Example of formal identification systems include the Uniform Resource Identifier (URI) (including the Uniform Resource Locator (URL)), the Digital Object Identifier (DOI) and the International Standard Book Number (ISBN).
DC-Lib Comments	*Standard identifier*: Provide at least one standard identifier from a standard scheme (e.g., URL, ISBN, etc.) if one or more standard identifiers have been assigned to the resource and are known to the metadata agency. *Unique-resource identifier*: Provide applicable identifiers assigned to one-of-a-kind resources (such as accession numbers assigned to items in a museum collection) if one or more of this class of identifiers have been assigned to the resource and are known to the metadata agency. If the structure of the identifier (e.g., all numerals) is potentially ambiguous it is recommended that the associated agencies' name be included as part of the identifier element. *Citation*: Provide a citation if no standard identifier is assigned and a formal citation is a common means of identifying the resource being described (e.g., a journal article). Use the element Identifier on a more abstract level; identifier for local library holdings like call number could be put into the DC-Lib element Location. (OpenURL may be registered as an encoding scheme).
Type of term	element
Refines	
Refined By	bibliographicCitation
Has Encoding Scheme	URI - http://purl.org/dc/terms/URI If not expressed by URI - SICI (Serial Item and Contribution Identifier), ISBN (International Standard Book Number), ISSN (International Standard Serial Number), DOI (Digital Object Identifier). To be registered as encoding schemes, URIs will be provided when available. Note: SICI and DOI may be registered as "info" URI schemes.
Obligation	**MA**
Occurence	

Name of Term	**bibliographicCitation**
Term URI	http://purl.org/dc/terms/bibliographicCitation
Label	Bibliographic Citation
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	A bibliographic reference for the resource.
DC-Lib Definition	Bibliographic citation information for a journal article, or similar bibliographic resource
Source Comments	Recommended practice is to include sufficient bibliographic detail to identify the resource as unambiguously as possible, whether or not the citation is in a standard form.

DC-Lib Comments	A draft version of "Guidelines for encoding bibliographic citations in DC metadata" can be found at http://epub.mimas.ac.uk/DC/dc-citation-guidelines/.
Type of term	element refinement
Refines	identifier
Refined By	
Has Encoding Scheme	
Obligation	**O**
Occurence	

Back to TOC

Name of Term	**source**
Term URI	http://purl.org/dc/elements/1.1/source
Label	Source
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	A Reference to a resource from which the present resource is derived.
DC-Lib Definition	
Source Comments	The present resource may be derived from the Source resource in whole or in part. Recommended best practice is to reference the Source resource by means of a string or number conforming to a formal identification system.
DC-Lib Comments	A constructed or derived ID of a local nature may be supplied, including a specification of the supplying organisation, in the absence of a globally unique one. Use only when the described resource is the result of digitization of non-digital originals. Otherwise, use Relation.
Type of term	element
Refines	
Refined By	
Has Encoding Scheme	URI - http://purl.org/dc/terms/URI SICI (Serial Item and Contribution Identifier), ISBN (International Standard Book Number), ISSN (International Standard Serial Number), DOI (Digital Object Identifier). To be registered as encoding schemes (these may also be expressed as URIs). URI will be provided when available. In the absence of a globally unique Identifier, a constructed or derived one of a local nature can be supplied, including a specification of the supplying organisation. (Consider registering OpenURL as an encoding scheme).
Obligation	**O**
Occurence	

Back to TOC

Name of Term	**language**
Term URI	http://purl.org/dc/elements/1.1/language
Label	Language
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	A language of the intellectual content of the resource.
DC-Lib Definition	
Source Comments	Recommended best practice is to use RFC 3066 [RFC3066], which, in conjunction with ISO 639 [ISO639], defines two- and three-letter primary language tags with optional subtags. Examples include "en" or "eng" for English, "akk" for Akkadian, and "en-GB" for English used in the United Kingdom
DC-Lib Comments	Language code may be used as a value for the Language qualifier to any DCMES element. Recommend codes rather than text, taken from ISO 639-2 bibliographic codes. Mandatory if applicable means if there is any spoken or written text, supply.
Type of term	element
Refines	
Refined By	

Has Encoding Scheme	ISO639-2 - DCMI approved encoding scheme. Use of the ISO 639-2 bibliographic code is preferred. A mapping is available at http://lcweb.loc.gov/standards/iso639-2/englangn.html. RFC 1766 - This scheme has been replaced by RFC 3066, which allows for a code from ISO 639-2 when there is no corresponding ISO 639-1 code. RFC 3066 is being registered as a DCMI approved scheme. RFC 3066 - Internet RFC 3066 'Tags for the Identification of Languages' specifies a primary subtag which is a two-letter code taken from ISO 639 part 1 or a three-letter code taken from ISO 639 part 2, followed optionally by a two-letter country code taken from ISO 3166. When a language in ISO 639 has both a two-letter and three-letter code, use the two-letter code; when it has only a three-letter code, use the three-letter code. This RFC replaces RFC 1766.
Obligation	**MA** (ISO 639-2), **O** (RFC 1766, RFC 3066)
Occurence	

Name of Term	**relation**
Term URI	http://purl.org/dc/elements/1.1/relation
Label	Relation
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	A reference to a related resource.
DC-Lib Definition	
Source Comments	Recommended best practice is to reference the resource by means of a string or number conforming to a formal identification system.
DC-Lib Comments	Relation without qualifier is optional; where it is qualified the recommendation for the qualifier should be followed. If using qualifiers, use the most specific one that is applicable. When no Identifier is available, a bibliographic description may be constructed. Future work will involve developing guidelines. Recommended use with qualifiers in certain situations: - When documents in hand are parts of "host documents" (e.g. journal, monographic series) and when there is no citation information in DC identifier. - When documents in hand are revisions or reformatted issues of earlier publications and information on these are readily available. OpenURL may be registered as an encoding scheme.
Type of term	element
Refines	
Refined By	isVersionOf, isFormatOf, hasFormat, isReplacedBy, Replaces, isPartOf, hasPart, Requires, isReferencedBy, References
Has Encoding Scheme	URI - http://purl.org/dc/terms/URI If not expressed by URI - SICI (Serial Item and Contribution Identifier), ISBN (International Standard Book Number), ISSN (International Standard Serial Number), DOI (Digital Object Identifier). To be registered as encoding schemes (these may also be expressed as URIs). http://dublincore.org/usage/terms/dc/current-schemes/
Obligation	**O**
Occurence	

Name of Term	**isVersionOf**
Term URI	http://purl.org/dc/terms/isVersionOf
Label	Is Version Of
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	The described resource is a version, edition, or adaptation of the referenced resource. Changes in version implies substantive changes in content rather than differences in format.
DC-Lib Definition	
Source Comments	
DC-Lib Comments	Do not include qualifier HasVersion, since this implies that it is clear which came first. Future work includes possibly identifying a need for HasVersion.
Type of term	element refinement
Refines	Relation
Refined By	

Has Encoding Scheme	URI - http://purl.org/dc/terms/URI If not expressed by URI - SICI (Serial Item and Contribution Identifier), ISBN (International Standard Book Number), ISSN (International Standard Serial Number), DOI (Digital Object Identifier). To be registered as encoding schemes (these may also be expressed as URIs). http://dublincore.org/usage/terms/dc/current-schemes/
Obligation	**R**
Occurence	

Name of Term	**isFormatOf**
Term URI	http://purl.org/dc/terms/isFormatOf
Label	Is Format Of
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	The described resource is the same intellectual content of the referenced resource, but presented in another format.
DC-Lib Definition	
Source Comments	
DC-Lib Comments	Use when there are alternative formats and it is not clear which preceded the other.
Type of term	element refinement
Refines	Relation
Refined By	
Has Encoding Scheme	URI - http://purl.org/dc/terms/URI If not expressed by URI - SICI (Serial Item and Contribution Identifier), ISBN (International Standard Book Number), ISSN (International Standard Serial Number), DOI (Digital Object Identifier). To be registered as encoding schemes (these may also be expressed as URIs). http://dublincore.org/usage/terms/dc/current-schemes/
Obligation	**R**
Occurence	

Name of Term	**hasFormat**
Term URI	http://purl.org/dc/terms/hasFormat
Label	Has Format
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	The described resource pre-existed the referenced resource, which is essentially the same intellectual content presented in another format.
DC-Lib Definition	
Source Comments	
DC-Lib Comments	Use only when the described resource references alternative physical formats. Example is the metadata for a home page for a dissertation that references that dissertation in various alternative formats (e.g. PDF, Postscript, etc.)
Type of term	element refinement
Refines	Relation
Refined By	
Has Encoding Scheme	URI - http://purl.org/dc/terms/URI If not expressed by URI - SICI (Serial Item and Contribution Identifier), ISBN (International Standard Book Number), ISSN (International Standard Serial Number), DOI (Digital Object Identifier). To be registered as encoding schemes (these may also be expressed as URIs). http://dublincore.org/usage/terms/dc/current-schemes/
Obligation	**R**
Occurence	

Name of Term	**isReplacedBy**
Term URI	http://purl.org/dc/terms/isReplacedBy
Label	Is Replaced By
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	The described resource is supplanted, displaced, or superceded by the referenced resource.
DC-Lib Definition	
Source Comments	
DC-Lib Comments	Used for succeeding version.
Type of term	element refinement
Refines	Relation
Refined By	

Has Encoding Scheme	URI If not expressed by URI - SICI (Serial Item and Contribution Identifier), ISBN (International Standard Book Number), ISSN (International Standard Serial Number), DOI (Digital Object Identifier). To be registered as encoding schemes (these may also be expressed as URIs). http://dublincore.org/usage/terms/dc/current-schemes/
Obligation	**O**
Occurence	

Name of Term	**replaces**
Term URI	http://purl.org/dc/terms/replaces
Label	Replaces
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	The described resource supplants, displaces, or supersedes the referenced resource.
DC-Lib Definition	
Source Comments	
DC-Lib Comments	Used for preceding version.
Type of term	element refinement
Refines	Relation
Refined By	
Has Encoding Scheme	URI - http://purl.org/dc/terms/URI If not expressed by URI - SICI (Serial Item and Contribution Identifier), ISBN (International Standard Book Number), ISSN (International Standard Serial Number), DOI (Digital Object Identifier). To be registered as encoding schemes (these may also be expressed as URIs). http://dublincore.org/usage/terms/dc/current-schemes/
Obligation	**O**
Occurence	

Name of Term	**isPartOf**
Term URI	http://purl.org/dc/terms/isPartOf
Label	Is Part Of
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	The described resource is a physical or logical part of the referenced resource.
DC-Lib Definition	
Source Comments	
DC-Lib Comments	Recommended use when documents in hand are parts of "host documents" (e.g. journal, monographic series) and when there is no citation information in DC identifier (if used by Citation WG).
Type of term	element refinement
Refines	Relation
Refined By	
Has Encoding Scheme	URI - http://purl.org/dc/terms/URI If not expressed by URI - SICI (Serial Item and Contribution Identifier), ISBN (International Standard Book Number), ISSN (International Standard Serial Number), DOI (Digital Object Identifier). To be registered as encoding schemes (these may also be expressed as URIs). http://dublincore.org/usage/terms/dc/current-schemes/
Obligation	**R**
Occurence	

Name of Term	**hasPart**
Term URI	http://purl.org/dc/terms/hasPart
Label	Has Part
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	The described resource includes the referenced resource either physically or logically.
DC-Lib Definition	
Source Comments	
DC-Lib Comments	
Type of term	element refinement
Refines	Relation
Refined By	
Has Encoding Scheme	URI - http://purl.org/dc/terms/URI If not expressed by URI - SICI (Serial Item and Contribution Identifier), ISBN (International Standard Book Number), ISSN (International Standard Serial Number), DOI (Digital Object Identifier). To be registered as encoding schemes (these may also be expressed as URIs). http://dublincore.org/usage/terms/dc/current-schemes/

Obligation	**O**
Occurence	

Name of Term	**requires**
Term URI	http://purl.org/dc/terms/requires
Label	Requires
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	The described resource requires the referenced resource to support its function, delivery, or coherence of content.
DC-Lib Definition	
Source Comments	
DC-Lib Comments	
Type of term	element refinement
Refines	Relation
Refined By	
Has Encoding Scheme	URI - http://purl.org/dc/terms/URI If not expressed by URI - SICI (Serial Item and Contribution Identifier), ISBN (International Standard Book Number), ISSN (International Standard Serial Number), DOI (Digital Object Identifier). To be registered as encoding schemes (these may also be expressed as URIs). http://dublincore.org/usage/terms/dc/current-schemes/
Obligation	**R**
Occurence	

Name of Term	**isReferencedBy**
Term URI	http://purl.org/dc/terms/isReferencedBy
Label	Is Referenced By
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	The described resource is referenced, cited, or otherwise pointed to by the referenced resource.
DC-Lib Definition	
Source Comments	
DC-Lib Comments	May be of limited use in terms of later resources referencing the initial resource, but may be useful to link to a major review or assessing essay.
Type of term	element refinement
Refines	Relation
Refined By	
Has Encoding Scheme	URI - http://purl.org/dc/terms/URI If not expressed by URI - SICI (Serial Item and Contribution Identifier), ISBN (International Standard Book Number), ISSN (International Standard Serial Number), DOI (Digital Object Identifier). To be registered as encoding schemes (these may also be expressed as URIs). http://dublincore.org/usage/terms/dc/current-schemes/
Obligation	**O**
Occurence	

Name of Term	**references**
Term URI	http://purl.org/dc/terms/references
Label	References
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	The described resource references, cites, or otherwise points to the referenced resource.
DC-Lib Definition	
Source Comments	
DC-Lib Comments	May be of limited use except for other resources that are far reaching or thorough criticisms. Not appropriate, for example, to include all references from the bibliography of the described resource in repeated Relation. References tags
Type of term	element refinement
Refines	Relation
Refined By	
Has Encoding Scheme	URI - http://purl.org/dc/terms/URI If not expressed by URI - SICI (Serial Item and Contribution Identifier), ISBN (International Standard Book Number), ISSN (International Standard Serial Number), DOI (Digital Object Identifier). To be registered as encoding schemes (these may also be expressed as URIs). http://dublincore.org/usage/terms/dc/current-schemes/
Obligation	**O**
Occurence	

Back to TOC

Name of Term	**coverage**
Term URI	http://purl.org/dc/elements/1.1/coverage
Label	Coverage
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	The extent or scope of the content of the resource
DC-Lib Definition	
Source Comments	Coverage will typically include spatial location (a place name or geographic coordinates), temporal period (a period label, date, or date range) or jurisdiction (such as a named administrative entity). Recommended best practice is to select a value from a controlled vocabulary (for example, the Thesaurus of Geographic Names [TGN]) and that, where appropriate, named places or time periods be used in preference to numeric identifiers such as sets of coordinates or date ranges.
DC-Lib Comments	Use Coverage with qualifier Spatial or Temporal; use of unqualified Coverage is discouraged in qualified DC.
Type of term	element
Refines	
Refined By	Spatial, Temporal
Has Encoding Scheme	See Source Comments above.
Obligation	**O**
Occurence	

Name of Term	**spatial**
Term URI	http://purl.org/dc/terms/spatial
Label	Spatial
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	Spatial characteristics of the intellectual content of the resource.
DC-Lib Definition	
Source Comments	Coverage will typically include spatial location (a place name or geographic coordinates), temporal period (a period label, date, or date range) or jurisdiction (such as a named administrative entity). Recommended best practice is to select a value from a controlled vocabulary (for example, the Thesaurus of Geographic Names [TGN]) and that, where appropriate, named places or time periods be used in preference to numeric identifiers such as sets of coordinates or date ranges.
DC-Lib Comments	Use this element for geographic coverage. The value of this element may also be included in Subject if desired. There is a need to evaluate DCMI Box and DCMI Point as for their usefulness for libraries and how they relate to current library practices for recording cartographic data.
Type of term	element refinement
Refines	Coverage
Refined By	
Has Encoding Scheme	DCMI Point - http://purl.org/dc/terms/Point ISO 3166 - http://purl.org/dc/terms/ISO3166 DCMI Box - http://purl.org/dc/terms/Box TGN - http://purl.org/dc/terms/TGN Use Library of Congress URI for MARC Geographic Area Codes, MARC Country Codes e.g. http://www.loc.gov/MARC.GAC
Obligation	**MA**
Occurence	

Name of Term	**temporal**
Term URI	http://purl.org/dc/terms/temporal
Label	Temporal
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	Temporal characteristics of the intellectual content of the resource.
DC-Lib Definition	
Source Comments	Coverage will typically include temporal period. Recommended best practice is to select a value from a controlled vocabulary.
DC-Lib Comments	The value of this element may also be included in Subject if desired. Prefer standard representation of date/time values in both DC.Date and dc.coverage.temporal, although textual descriptions may also be used.
Type of term	element refinement
Refines	Coverage
Refined By	
Has Encoding Scheme	DCMI Period - http://purl.org/dc/terms/Period W3C-DTF - http://purl.org/dc/terms/W3CDTF
Obligation	**MA**

Occurence	

Back to TOC

Name of Term	**rights**
Term URI	http://purl.org/dc/elements/1.1/rights
Label	Rights
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	Information about rights held in and over the resource.
DC-Lib Definition	
Source Comments	Typically, a Rights element will contain a rights management statement for the resource, or reference a service providing such information. Rights information often encompasses Intellectual Property Rights (IPR), Copyright, and various Property Rights. If the Rights element is absent, no assumptions can be made about the status of these and other rights with respect to the resource.
DC-Lib Comments	Need to determine how to use for library applications; there is ongoing discussion on rights metadata in various applications.
Type of term	element
Refines	
Refined By	
Has Encoding Scheme	URI - http://purl.org/dc/terms/URI
Obligation	**R if applicable (if there are encumbrances)**
Occurence	

Back to TOC

Name of Term	**audience**
Term URI	http://purl.org/dc/terms/audience
Label	Audience
Defined By	http://dublincore.org/documents/dcmi-terms/
Source Definition	A class of entity for whom the resource is intended or useful.
DC-Lib Definition	
Source Comments	A class of entity may be determined by the creator or the publisher or by a third party.
DC-Lib Comments	Approved by DCMI Usage Board October 2001. Need to evaluate whether any refinements or encoding scheme(s) are appropriate for DC-Lib use. MARC target audience codes may be considered.
Type of term	element
Refines	
Refined By	
Has Encoding Scheme	See DC-Lib comment
Obligation	**O**
Occurence	

Back to TOC

Name of Term	**edition**
Term URI	http://www.loc.gov/mods
Label	Edition
Defined By	
Source Definition	
DC-Lib Definition	Information designating the version or edition of a work.
Source Comments	
DC-Lib Comments	Being able to specify the version or edition of a given work is often critical to successful resource discovery and identification to determine whether a resource is the same as another one. This is particularly important for resources that change frequently. This is not to be used for versions in the sense of different physical formats (e.g. the PDF version of a textual resource). Use existing element <edition> under <publicationInfo> in the Metadata Object Description Schema Decision by DCMI Usage Board in May 2002. This element will generally not be repeated. Element should be included if necessary for identification.
Type of term	element
Refines	
Refined By	
Has Encoding Scheme	
Obligation	**R**
Occurence	

Occurence	

Name of Term	**location**
Term URI	http://www.loc.gov/mods
Label	Location
Defined By	
Source Definition	
DC-Lib Definition	Identifies the organization holding the resource or from which access is obtained.
Source Comments	
DC-Lib Comments	Use for a physical location that allows the user to retrieve the item when a URI is not appropriate (e.g. for physical items not available electronically). This also facilitates access if the URI doesn't retrieve anything or only a poor substitute. Can also contain further identification within a location such as call number, accession number. Use existing element <location> in the Metadata Object Description Schema Decision by DCMI Usage Board in May 2002.
Type of term	element
Refines	
Refined By	
Has Encoding Scheme	MARC Code list for Organizations - http://www.loc.gov/marc/organizations/ Need to evaluate other encoding schemes, including ISO/DIS 15511.3 (International Standard Identifier for Libraries and Related Organizations (ISIL)). May also want to investigate a URI to an entry in an authority file.
Obligation	**MA**
Occurence	

Back to TOC

Encoding Schemes

Name of Term	**Box**
Term URI	http://purl.org/dc/terms/Box
Label	DCMI Box
Defined By	http://purl.org/dc/terms/
Definition	The DCMI Box identifies a region of space using its geographic limits.
Comments	
See Also	http://dublincore.org/documents/dcmi-box/
Type of term	encoding scheme
Encoding Scheme For	Spatial

Name of Term	**DCMIType**
Term URI	http://purl.org/dc/dcmitype/
Label	DCMI Type Vocabulary
Defined By	http://purl.org/dc/terms/
Definition	A list of types used to categorize the nature or genre of the content of the resource.
Comments	
See Also	http://dublincore.org/documents/dcmi-type-vocabulary/
Type of term	encoding scheme
Encoding Scheme For	Type

Name of Term	**DDC**
Term URI	http://purl.org/dc/terms/DDC
Label	DDC
Defined By	http://purl.org/dc/terms/
Definition	Dewey Decimal Classification
Comments	
See Also	http://www.oclc.org/dewey/
Type of term	encoding scheme
Encoding Scheme For	Subject

Name of Term	**DOI**
Term URI	URI will be provided when available
Label	DOI

Defined By	
Definition	Digital Object Identifier
Comments	May be registered as "info" URI scheme.
See Also	http://www.doi.org/
Type of term	encoding scheme
Encoding Scheme For	Identifier, Source, IsVersionOf, IsFormatOf, HasFormat, IsReplacedBy, Replaces, IsPartOf, HasPart, Requires, IsReferencedBy, References

Name of Term	**IMT**
Term URI	http://purl.org/dc/terms/IMT
Label	IMT
Defined By	http://purl.org/dc/terms/
Definition	The Internet media type of the resource.
Comments	
See Also	http://www.isi.edu/in-notes/iana/assignments/media-types/media-types
Type of term	encoding scheme
Encoding Scheme For	Format, Medium

Back to TOC

Name of Term	**ISBN**
Term URI	URI will be provided when available
Label	ISBN
Defined By	
Definition	International Standard Book Number
Comments	
See Also	http://www.isbn.org/standards/home/isbn/international/index.asp
Type of term	encoding scheme
Encoding Scheme For	Identifier, Source, IsVersionOf, IsFormatOf, HasFormat, IsReplacedBy, Replaces, IsPartOf, HasPart, Requires, IsReferencedBy, References

Name of Term	**ISO3166**
Term URI	http://purl.org/dc/terms/ISO3166
Label	ISO 3166
Defined By	http://purl.org/dc/terms/
Definition	ISO 3166 Codes for the representation of names of countries.
Comments	
See Also	http://www.din.de/gremien/nas/nabd/iso3166ma/codlstp1/index.html
Type of term	encoding scheme
Encoding Scheme For	Spatial

Name of Term	**ISO639-2**
Term URI	http://purl.org/dc/terms/ISO639-2
Label	ISO 639-2
Defined By	http://purl.org/dc/terms/
Definition	ISO 639-2: Codes for the representation of names of languages
Comments	
See Also	http://www.loc.gov/standards/iso639-2/
Type of term	encoding scheme
Encoding Scheme For	Language

Name of Term	**ISO8601**
Term URI	http://purl.org/dc/terms/ISO8601
Label	ISO 8601
Defined By	http://purl.org/dc/terms/
Definition	
Comments	
See Also	
Type of term	encoding scheme
Encoding Scheme For	Date, Created, Valid, Available, Issued, Modified, DateCopyrighted, DateSubmitted, DateAccepted, DateCaptured

Name of Term	**ISSN**
Term URI	URI will be provided when available

Label	ISSN
Defined By	
Definition	International Standard Serial Number
Comments	
See Also	http://www.issn.org:8080/pub/
Type of term	encoding scheme
Encoding Scheme For	Identifier, Source, IsVersionOf, IsFormatOf, HasFormat, IsReplacedBy, Replaces, IsPartOf, HasPart, Requires, IsReferencedBy, References

Name of Term	**LCC**
Term URI	http://purl.org/dc/terms/LCC
Label	LCC
Defined By	http://purl.org/dc/terms/
Definition	Library of Congress Classification
Comments	
See Also	http://lcweb.loc.gov/catdir/cpso/lcco/lcco.html
Type of term	encoding scheme
Encoding Scheme For	Subject

Name of Term	**LCSH**
Term URI	http://purl.org/dc/terms/LCSH
Label	LCSH
Defined By	http://purl.org/dc/terms/
Definition	Library of Congress Subject Headings
Comments	
See Also	http://lcweb.loc.gov/cds/lcsh.html
Type of term	encoding scheme
Encoding Scheme For	Subject

Name of Term	**MESH**
Term URI	http://purl.org/dc/terms/MESH
Label	MESH
Defined By	http://purl.org/dc/terms/
Definition	Medical Subject Headings
Comments	
See Also	http://nlm.nih.gov/mesh/meshhome.html
Type of term	encoding scheme
Encoding Scheme For	Subject

Back to TOC

Name of Term	**Period**
Term URI	http://purl.org/dc/terms/Period
Label	DCMI Period
Defined By	http://purl.org/dc/terms/
Definition	A specification of the limits of a time interval.
Comments	
See Also	http://dublincore.org/documents/dcmi-period/
Type of term	encoding scheme
Encoding Scheme For	Temporal

Name of Term	**Point**
Term URI	http://purl.org/dc/terms/Point
Label	DCMI Point
Defined By	http://purl.org/dc/terms/
Definition	The DCMI Point identifies a point in space using its geographic coordinates.
Comments	
See Also	http://dublincore.org/documents/dcmi-point/
Type of term	encoding scheme
Encoding Scheme For	Spatial

Name of Term	**RFC3066**
Term URI	http://purl.org/dc/terms/RFC3066

Label	RFC3066
Defined By	http://purl.org/dc/terms/
Definition	Internet RFC 3066 'Tags for the Identification of Languages' specifies a primary subtag which is a two-letter code taken from ISO 639 part 1 or a three-letter code taken from ISO 639 part 2, followed optionally by a two-letter country code taken from ISO 3166. When a language in ISO 639 has both a two-letter and three-letter code, use the two-letter code; when it has only a three-letter code, use the three-letter code. This RFC replaces RFC 1766.
Comments	
See Also	http://www.ietf.org/rfc/rfc3066.txt
Type of term	encoding scheme
Encoding Scheme For	Language

Name of Term	**SICI**
Term URI	URI will be provided when available
Label	SICI
Defined By	
Definition	Serial Item and Contributor Identifier
Comments	May be registered as "info" URI scheme.
See Also	http://sunsite.berkeley.edu/SICI/
Type of term	encoding scheme
Encoding Scheme For	Identifier, Source, IsVersionOf, IsFormatOf, HasFormat, IsReplacedBy, Replaces, IsPartOf, HasPart, Requires, IsReferencedBy, References

Name of Term	**TGN**
Term URI	http://purl.org/dc/terms/TGN
Label	TGN
Defined By	http://purl.org/dc/terms/
Definition	The Getty Thesaurus of Geographic Names.
Comments	
See Also	http://www.getty.edu/research/tools/vocabulary/tgn/index.html
Type of term	encoding scheme
Encoding Scheme For	Spatial

Name of Term	**UDC**
Term URI	http://purl.org/dc/terms/UDC
Label	UDC
Defined By	http://purl.org/dc/terms/
Definition	Universal Decimal Classification
Comments	
See Also	http://www.udcc.org/
Type of term	encoding scheme
Encoding Scheme For	Subject

Name of Term	**URI**
Term URI	http://purl.org/dc/terms/URI
Label	URI
Defined By	http://purl.org/dc/terms/
Definition	A URI uniform resource identifier.
Comments	
See Also	http://www.itef.org/rfc/rfc2396.txt
Type of term	encoding scheme
Encoding Scheme For	Description, Identifier, Source, Relation, Rights, Abstract, TableOfContents, IsVersionOf, IsFormatOf, HasFormat, IsReplacedBy, Replaces, IsPartOf, HasPart, Requires, IsReferencedBy, References

Name of Term	**W3CDTF**
Term URI	http://purl.org/dc/terms/W3CDTF
Label	W3C-DTF
Defined By	http://purl.org/dc/terms/
Definition	W3C encoding rules for dates and times - a profile based on ISO 8601.
Comments	
See Also	http://www.w3c.org/TR/NOTE-datetime
Type of term	encoding scheme
Encoding Scheme For	Date, Created, Valid, Available, Issued, Modified, DateCopyrighted, DateSubmitted, DateAccepted, DateCaptured, Temporal

Back to TOC

5. Acknowledgements

Thanks to all members of the DC-Libraries Application Profile drafting committee who participated in the first draft of this application profile. Thanks to Nicki Clegg of Web Services Delivery Unit, the British Library, for her work in re-formatting the profile. This revision is based on decisions made by the Usage Board in 2003 and Working Group discussion.

Members of the working group:

Ann Apps (University of Manchester)
Olga Barysheva (National Library of Russia)
Diane Boehr (National Library of Medicine)
Priscilla Caplan (Florida Center for Library Automation
Warwick Cathro (National Library of Australia)
Ann Chapman (UKOLN)
Hsueh-hua Chen (National Taiwan University)
Eric Childress (OCLC)
Robina Clayphan (British Library)
Monika Cremer (University of Goettingen)
Stina Degerstedt (Koniglige Bibliotek, Sweden)
Ricky Erway (Research Libraries Group)
Carolyn Guinchard (University of Alberta)
Rebecca Guenther (Library of Congress)
Susan Haigh (National Library of Canada)
Sten Hedberg (Uppsala University Library)
Rachel Heery (UKOLN)
Christel Hengel (Deutsch Bibliothek)
Diane Hillmann (Cornell University
Noriko Kando (NII)
Wei Liu (Shanghai Library)
Lynn Marko (University of Michigan)
Heike Neuroth (University of Goettingen)
Trudi Noordermeer (Royal Library of the Netherlands)
Marianne Peereboom (Royal Library of the Netherlands)
Shigeo Sugimoto (University of Library and Information Science, Japan)
Stuart Weibel (DCMI)
Robin Wendler (Harvard University)
Mary Woodley (California State University, Northridge)

6. Major changes since last update

- Reformatted to conform to the CEN DC AP Guidelines.
- "Best practice" and "Open questions" from previous versionmoved to "DC-Lib comments".
- Encoding schemes referenced in table of relevant element and expanded in separate tables at the end of the profile.
- Date, type, identifier, and coverage elements edited to reflect Usage Board decisions.
- DC URI added for ISO8601 in anticipation of this being included in dmi-terms document.
- Creator, contributor, and publisher updated to reflect Role refinement decision.
- Other edits according to Working Group discussions see http://www.jiscmail.ac.uk/cgi-bin/webadmin?A2=ind0311&L=dc-libraries&T=0&F=&S=&P=153

Metadata associated with this resource: http://dublincore.org/documents/2004/09/10/library-application-profile/index.shtml.rdf

Title: # DCMI Type Vocabulary - draft 2005-05-11

Creator:	DCMI Usage Board
Identifier:	http://dublincore.org/documents/2003/11/19/dcmi-type-vocabulary/
Date Issued:	2003-11-19
Latest Version:	http://dublincore.org/documents/dcmi-type-vocabulary/
Replaces:	http://dublincore.org/documents/2003/02/12/dcmi-type-vocabulary/
Replaced By:	Not applicable
Translations:	http://dublincore.org/resources/translations/
Document Status:	This is a DCMI Recommendation.
Description:	The DCMI Type Vocabulary provides a general, cross-domain list of approved terms that may be used as values for the Resource Type element to identify the genre of a resource. The terms documented here are also included in the more comprehensive document "DCMI Metadata Terms" at http://dublincore.org/documents/dcmi-terms/.
Date Valid:	2003-11-19

Term Name: Collection

URI:	http://purl.org/dc/dcmitype/Collection
Label:	Collection
Definition:	An aggregation of items.
Comment:	The term collection means that the resource is described as a group; its parts may be separately described and navigated.
Type of Term:	vocabulary-term
Status:	recommended
Date Issued:	2000-07-11
Version:	Collection-001

Term Name: Dataset

URI:	http://purl.org/dc/dcmitype/Dataset
Label:	Dataset
Definition:	Information encoded in a defined structure (for example, lists, tables, and databases).
Comment:	A dataset is intended to be useful for direct machine processing.
Type of Term:	vocabulary-term
Status:	recommended
Date Issued:	2000-07-11
Version:	Dataset-001

Term Name: Event

URI:	http://purl.org/dc/dcmitype/Event
Label:	Event
Definition:	A non-persistent, time-based occurrence.
Comment:	Metadata for an event provides descriptive information that is the basis for discovery of the purpose, location, duration, responsible agents, and links to related events and resources. The resource of type event may not be retrievable if the described instantiation has expired or is yet to occur. Examples - exhibition, web-cast, conference, workshop, open-day, performance, battle, trial, wedding, tea-party, conflagration.

Type of Term:	vocabulary-term
Status:	recommended
Date Issued:	2000-07-11
Version:	Event-001

Term Name: Image

URI:	http://purl.org/dc/dcmitype/Image
Label:	Image
Definition:	A primarily symbolic visual representation other than text.
Comment:	For example - images and photographs of physical objects, paintings, prints, drawings, other images and graphics, animations and moving pictures, film, diagrams, maps, musical notation. Note that image may include both electronic and physical representations.
Type of Term:	vocabulary-term
Broader Than:	http://purl.org/dc/dcmitype/StillImage
Broader Than:	http://purl.org/dc/dcmitype/MovingImage
Status:	recommended
Date Issued:	2000-07-11
Version:	Image-002

Term Name: InteractiveResource

URI:	http://purl.org/dc/dcmitype/InteractiveResource
Label:	Interactive Resource
Definition:	A resource which requires interaction from the user to be understood, executed, or experienced.
Comment:	For example - forms on web pages, applets, multimedia learning objects, chat services, virtual reality.
Type of Term:	vocabulary-term
Status:	recommended
Date Issued:	2000-07-11
Version:	InteractiveResource-001

Term Name: Service

URI:	http://purl.org/dc/dcmitype/Service
Label:	Service
Definition:	A system that provides one or more functions of value to the end-user.
Comment:	Examples include: a photocopying service, a banking service, an authentication service, interlibrary loans, a Z39.50 or Web server.
Type of Term:	vocabulary-term
Status:	recommended
Date Issued:	2000-07-11
Version:	Service-001

Term Name: Software

URI:	http://purl.org/dc/dcmitype/Software

Label:	Software
Definition:	A computer program in source or compiled form which may be available for installation non-transiently on another machine.
Comment:	For software which exists only to create an interactive environment, use interactive instead.
Type of Term:	vocabulary-term
Status:	recommended
Date Issued:	2000-07-11
Version:	Software-001

Term Name: Sound

URI:	http://purl.org/dc/dcmitype/Sound
Label:	Sound
Definition:	A resource whose content is primarily intended to be rendered as audio.
Comment:	For example - a music playback file format, an audio compact disc, and recorded speech or sounds.
Type of Term:	vocabulary-term
Status:	recommended
Date Issued:	2000-07-11
Version:	Sound-001

Term Name: Text

URI:	http://purl.org/dc/dcmitype/Text
Label:	Text
Definition:	A resource whose content is primarily words for reading.
Comment:	For example - books, letters, dissertations, poems, newspapers, articles, archives of mailing lists. Note that facsimiles or images of texts are still of the genre text.
Type of Term:	vocabulary-term
Status:	recommended
Date Issued:	2000-07-11
Version:	Text-001

Term Name: PhysicalObject

URI:	http://purl.org/dc/dcmitype/PhysicalObject
Label:	Physical Object
Definition:	An inanimate, three-dimensional object or substance.
Comment:	Note that digital representations of, or surrogates for, these things should use Image, Text or one of the other types.
Type of Term:	vocabulary-term
Status:	recommended
Date Issued:	2002-07-13
Version:	PhysicalObject-001

Term Name: StillImage

URI:	http://purl.org/dc/dcmitype/StillImage

Label:	Still Image
Definition:	A static visual representation.
Comment:	Examples of still images are: paintings, drawings, graphic designs, plans and maps. Recommended best practice is to assign the type "text" to images of textual materials. Instances of the type "Still Image" must also be describable as instances of the broader type "Image".
Type of Term:	vocabulary-term
Narrower Than:	http://purl.org/dc/dcmitype/Image
Status:	recommended
Date Issued:	2003-11-18
Version:	StillImage-001

Term Name: MovingImage

URI:	http://purl.org/dc/dcmitype/MovingImage
Label:	Moving Image
Definition:	A series of visual representations that, when shown in succession, impart an impression of motion.
Comment:	Examples of moving images are: animations, movies, television programs, videos, zoetropes, or visual output from a simulation. Instances of the type "Moving Image" must also be describable as instances of the broader type "Image".
Type of Term:	vocabulary-term
Narrower Than:	http://purl.org/dc/dcmitype/Image
Status:	recommended
Date Issued:	2003-11-18
Version:	MovingImage-001

Date: Mon, 2 May 2005 23:06:40 +0100
From: Andy Powell <a.powell@UKOLN.AC.UK>
Subject: Re: Edits to the type vocabulary

On Mon, 2 May 2005, Stuart Sutton wrote:
> All, some time back, Diane and I were charged with doing minor edits of the
> type vocabulary. My recollection was that we were to move best practice
> information etc. from the description to the comments. We did a first stab
> at those edits and the UB reviewed them. It was the sense of the UB that
> there were additional edits needing doing. Unfortunately, so much time has
> past that I cannot for the life of me figure out (or find information)
> regarding these additional edits. I have attached an HTML page here with
> the edits as they now stand. Can anyone jog my memory regarding what
> remains to be done?

I don't remember either I'm afraid.

Looking at your edited version now, I have the following comments.

I think that the comment for software is completely broken

```
Label:  Software
        Definition:     A computer program in source or compiled form
                        which may be available for installation
                        non-transiently on another machine.
        Comment:        For software which exists only to create an
                        interactive environment, use interactive instead.
```

The last sentence should at least say 'Interactive Resource'.
But the more fundamental problem is that the comment mixes
up descriptions of the 'software' from descriptions of the
'thing created by running the software', i.e. the 'interactive
resource'. In some situations it'll be appropriate to describe
one, in other situations it'll be approapriate to describe
the other - but you can't simply recommend to use one of these
in favour of the other - since they are different resources!

In the comment for Collection

```
        Label:  Collection
        Definition:     An aggregation of items.
        Comment:        The term collection means that the resource is
                        described as a group; its parts may be separately
                        described and navigated.
```

it might be helpful to say "its parts may *also* be separately
described and navigated"?

The comment for Event is very odd. An Event is *never*
retreivable! And why does this comment talk about the other
metadata about the event - none of the other comments do this.

The definition for Sound seems to mix up work and manifestation
(or reality and representation, or something) - i.e. a 'sound'
and a 'sound file'. The definition sounds like the latter,
but I suspect the former is also intended??

```
Label:  Sound
        Definition:     A resource whose content is primarily intended to
                        be rendered as audio.
        Comment:        For example - a music playback file format, an
                        audio compact disc, and recorded speech or sounds.
```

Can I use DC to describe birdsong? Or can I only use it to describe a
recording of birdsong? If the Sound type only applies to the latter

(which is what the definition sounds like it means) then what is the type
of the former?

Date: Wed, 4 May 2005 06:29:21 -0700
From: Stuart Sutton <sasutton@U.WASHINGTON.EDU>
Subject: Re: Edits to the type vocabulary

Thanks, Andy. I'm not quite sure how to approach your comments in the
course of these edits. Several go to the heart of terms semantics. It was
my understanding that what Diane and I were changed to do was rather
ministerial in terms of moving text more appropriate to the comments from
the current descriptions and to make the texts more consistent. In doing
the first round, we tried to be as loyal to the current semantics as we
could. Your points are well made; but I guess I am not sure whether Diane
and I should go off re-crafting semantics. Perhaps there might be a moment
on tomorrow's call to clarify the charge in light of your comments...or to
postpone discussion until D.C. in a few weeks.

From: Andy Powell <a.powell@UKOLN.AC.UK>
Subject: Re: Edits to the type vocabulary

On Wed, 4 May 2005, Stuart Sutton wrote:
> Thanks, Andy. I'm not quite sure how to approach your comments in the
> course of these edits. Several go to the heart of terms semantics. It was
> my understanding that what Diane and I were changed to do was rather
> ministerial in terms of moving text more appropriate to the comments from
> the current descriptions and to make the texts more consistent. In doing
> the first round, we tried to be as loyal to the current semantics as we
> could. Your points are well made; but I guess I am not sure whether Diane
> and I should go off re-crafting semantics.

Agreed. Fair point.

But I think that my comments about Software, Collection and Event fall
into the non-semantic category? The only really contentious comment (of
mine) is the one about sound??

DCMI Type Vocabulary

Title: DCMI Type Vocabulary
Creator: DCMI Usage Board
Identifier: http://dublincore.org/documents/2004/06/14/dcmi-type-vocabulary/
Date Issued: 2004-06-14
Latest Version: http://dublincore.org/documents/dcmi-type-vocabulary/
Replaces: http://dublincore.org/documents/2003/11/19/dcmi-type-vocabulary/
Replaced By: Not applicable
Translations: http://dublincore.org/resources/translations/
Document Status: This is a DCMI Recommendation.
Description: The DCMI Type Vocabulary provides a general, cross-domain list of approved terms that may be used as values for the Resource Type element to identify the genre of a resource. The terms documented here are also included in the more comprehensive document "DCMI Metadata Terms" at http://dublincore.org/documents/dcmi-terms/.
Date Valid: 2004-06-14

Term Name: Collection	
URI:	http://purl.org/dc/dcmitype/Collection
Label:	Collection
Definition:	A collection is an aggregation of items. The term collection means that the resource is described as a group; its parts may be separately described and navigated.
Type of Term:	vocabulary-term
Status:	recommended
Date Issued:	2000-07-11
Version:	Collection-001
Term Name: Dataset	
URI:	http://purl.org/dc/dcmitype/Dataset
Label:	Dataset
Definition:	A dataset is information encoded in a defined structure (for example, lists, tables, and databases), intended to be useful for direct machine processing.
Type of Term:	vocabulary-term
Status:	recommended
Date Issued:	2000-07-11
Version:	Dataset-001
Term Name: Event	
URI:	http://purl.org/dc/dcmitype/Event
Label:	Event
Definition:	An event is a non-persistent, time-based occurrence. Metadata for an event provides descriptive information that is the basis for discovery of the purpose, location, duration, responsible agents, and links to related events and resources. The resource of type event may not be retrievable if the described instantiation has expired or is yet to occur. Examples - exhibition, web-cast, conference, workshop, open-day, performance, battle, trial, wedding, tea-party, conflagration.

Type of Term:	vocabulary-term
Status:	recommended
Date Issued:	2000-07-11
Version:	Event-001
Term Name: Image	
URI:	http://purl.org/dc/dcmitype/Image
Label:	Image
Definition:	An image is a primarily symbolic visual representation other than text. For example - images and photographs of physical objects, paintings, prints, drawings, other images and graphics, animations and moving pictures, film, diagrams, maps, musical notation. Note that image may include both electronic and physical representations.
Type of Term:	vocabulary-term
Broader Than:	http://purl.org/dc/dcmitype/StillImage
Broader Than:	http://purl.org/dc/dcmitype/MovingImage
Status:	recommended
Date Issued:	2000-07-11
Version:	Image-002
Term Name: InteractiveResource	
URI:	http://purl.org/dc/dcmitype/InteractiveResource
Label:	Interactive Resource
Definition:	An interactive resource is a resource which requires interaction from the user to be understood, executed, or experienced. For example - forms on web pages, applets, multimedia learning objects, chat services, virtual reality.
Type of Term:	vocabulary-term
Status:	recommended
Date Issued:	2000-07-11
Version:	InteractiveResource-001
Term Name: MovingImage	
URI:	http://purl.org/dc/dcmitype/MovingImage
Label:	Moving Image
Definition:	A series of visual representations that, when shown in succession, impart an impression of motion. Examples of moving images are: animations, movies, television programs, videos, zoetropes, or visual output from a simulation.
Comment:	Instances of the type "Moving Image" must also be describable as instances of the broader type "Image".
Type of Term:	vocabulary-term
Narrower Than:	http://purl.org/dc/dcmitype/Image
Status:	recommended
Date Issued:	2003-11-18
Version:	MovingImage-001
Term Name: PhysicalObject	
URI:	http://purl.org/dc/dcmitype/PhysicalObject
Label:	Physical Object
Definition:	An inanimate, three-dimensional object or substance. For example -- a computer, the great pyramid, a sculpture. Note that digital representations of, or surrogates for, these things should use Image, Text or one of the other types.
Type of Term:	vocabulary-term
Status:	recommended
Date Issued:	2002-07-13
Version:	PhysicalObject-001
Term Name: Service	
URI:	http://purl.org/dc/dcmitype/Service
Label:	Service
Definition:	A service is a system that provides one or more functions of value to the end-user. Examples include: a photocopying service, a banking service, an authentication service, interlibrary loans, a Z39.50 or Web server.

Type of Term:	vocabulary-term
Status:	recommended
Date Issued:	2000-07-11
Version:	Service-001
Term Name: Software	
URI:	http://purl.org/dc/dcmitype/Software
Label:	Software
Definition:	Software is a computer program in source or compiled form which may be available for installation non-transiently on another machine. For software which exists only to create an interactive environment, use interactive instead.
Type of Term:	vocabulary-term
Status:	recommended
Date Issued:	2000-07-11
Version:	Software-001
Term Name: Sound	
URI:	http://purl.org/dc/dcmitype/Sound
Label:	Sound
Definition:	A sound is a resource whose content is primarily intended to be rendered as audio. For example - a music playback file format, an audio compact disc, and recorded speech or sounds.
Type of Term:	vocabulary-term
Status:	recommended
Date Issued:	2000-07-11
Version:	Sound-001
Term Name: StillImage	
URI:	http://purl.org/dc/dcmitype/StillImage
Label:	Still Image
Definition:	A static visual representation. Examples of still images are: paintings, drawings, graphic designs, plans and maps.
Comment:	Recommended best practice is to assign the type "text" to images of textual materials. Instances of the type "Still Image" must also be describable as instances of the broader type "Image".
Type of Term:	vocabulary-term
Narrower Than:	http://purl.org/dc/dcmitype/Image
Status:	recommended
Date Issued:	2003-11-18
Version:	StillImage-001
Term Name: Text	
URI:	http://purl.org/dc/dcmitype/Text
Label:	Text
Definition:	A text is a resource whose content is primarily words for reading. For example - books, letters, dissertations, poems, newspapers, articles, archives of mailing lists. Note that facsimiles or images of texts are still of the genre text.
Type of Term:	vocabulary-term
Status:	recommended
Date Issued:	2000-07-11
Version:	Text-001

Metadata associated with this resource: http://dublincore.org/documents/dcmi-type-vocabulary/index.shtml.rdf

DCMI Usage Board Administrative Processes

Creator:	Diane I. Hillmann
Creator:	Stuart A. Sutton
Date Issued:	@@@
Identifier:	@@@
Replaces:	@@@
Is Replaced By:	Not Applicable
Latest version:	http://dublincore.org/usage/documents/process/
Description of document:	This document describes the process by which the DCMI Usage Board reaches decisions on terms and application profiles, as well as its process for managing the registration of encoding schemes.

Index

Preface

The Usage Board acts in accordance with its charter under the DCMI Bylaws, Article II, section D. While providing more guiding detail then the broad mandate of the DCMI Bylaws, the following process statements are intended to comport fully with that mandate. Should discrepancies between these process statements and the DCMI Bylaws emerge, the DCMI Bylaws control.

The following process statements guide the Usage Board in executing it responsibility "to ensure an orderly evolution of the metadata terms maintained by the Dublin Core Metadata Initiative." The process statements are amended from time to time to reflect the evolving nature of DCMI and metadata.

Part 1-Usage Board: Overview, Meetings, Documentation

1. Usage Board Membership

1.1. The UB will consist of at least seven and no more than eleven people (nine is ideal) appointed by the DCMI Directorate.
1.2. Usage Board member terms shall be for two years, renewable once. Initial appointments will be made so as to stagger terms.
1.3. Members should be selected based on the following criteria:

1.3.1. Knowledgeable concerning the development history and purpose of the DC element set and its relationship to the metadata world at large;
1.3.2. Related to a metadata community relevant to DCMI;
1.3.3. Willing and able to commit time and energy to the functions of the UB;
1.3.4. Able to communicate verbally and in writing in English well enough to prepare documents and discuss complex issues in a group setting;
1.3.5. Geographic and domain distribution of members is relevant but will not override other criteria.

1.4. The UB Chair will be appointed from one of the membership by the DCMI Directorate. The term of the chair shall be for two years, renewable once.
1.5. Liaisons from DCMI affiliates may be appointed by DCMI management in consultation with the Usage Board Chair.

1.5.1. Liaisons are non-voting and do not serve as shepherds, but are encouraged to participate in discussion on the Usage Board list and at meetings.

1.6. For internal communication the UB uses the closed mailing list dc-usage@jiscmail.ac.uk. The messages are archived and publicly available at http://www.jiscmail.ac.uk/lists/dc-usage.html.

2. Meetings

2.1. Scheduling

2.1.1. Meetings should be held at least twice a year.

2.1.1.1. One meeting should be scheduled during the annual DC general workshop/conference.
2.1.1.2. The second should be scheduled at a different time of the year, preferably close to other conferences, so as to make attendance convenient for as many members as possible.
2.1.1.3. Scheduling should be done far enough in advance so that as many members as possible may be present.

2.2. Funding for regular UB members attendance at meetings should be supported as much as possible by DCMI.

2.2.1. Funding for the attendance of Liaisons at UB meetings should be supported by their institution.

2.3. Meeting agenda

2.3.1. The UB Chair maintains the agenda, which cites links to relevant supporting

documentation, including JISCMAIL postings.
2.3.2. All materials pointed to in the agenda are archived at http://dublincore.org/usage/meetings/ after the final pre-meeting version of the agenda has been distributed. After the meeting, the archive version of the agenda is edited to point to these archive copies.

2.4. Attendance by members

2.4.1. Members must attend at least one meeting in a given year to maintain membership in good standing.
2.4.2. Members who miss two meetings in succession may be replaced by the DC Directorate.

2.5. Attendance by others

2.5.1. Attendance at UB meetings by other than the UB is by invitation.

2.5.1.1. People interested in attending should request an invitation via the UB Chair or the Managing Director.

2.5.2. Participation in discussion of proposals by any interested parties is encouraged.

2.6. Agenda preparation and distribution

2.6.1. The UB chair is responsible for preparing the meeting agendas and assigning shepherds to proposals.
2.6.2. Agenda items shall include the name and email address of the UB member responsible for shepherding the proposal through the UB process.
2.6.3. Agendas shall be available at http://www.dublincore.org/usage/meetings/ a few weeks before the meeting.

2.7. Important decisions will be assigned a number for citation purposes and documented on the DCMI website.

3. Categories of Usage Board Decisions

3.1. ***Recommended*****:** Elements, Element Refinements, and DCMI-maintained Vocabulary Terms (e.g., member terms of the DCMI Type Vocabulary) useful for resource discovery across domains.
3.2. ***Conforming*****:** Elements, Element Refinements and Application Profiles may be assigned a status of conforming. Elements and Element Refinements assigned a status of conforming are those for which an implementation community has a demonstrated need and which conform to the grammar of Elements and Element Refinements, though without necessarily meeting the stricter criteria of usefulness across domains or usefulness for resource discovery.
3.3. ***Obsolete*****:** For Elements and Element Refinements that have been superseded, deprecated, or rendered obsolete. Such terms will remain in the registry for use in interpreting legacy metadata.
3.4. ***Registered*****:** Used for Vocabulary Encoding Schemes and language translations for which the DCMI provides information but not necessarily a specific recommendation.
3.5. ***Endorsed*****:** @@@

Part 2-Proposals: Form and Process

4. Proposals for Terms

4.1. Sources of proposals may be:

4.1.1. DCMI working groups

4.1.1.1. Existing working groups
4.1.1.2. Working groups established for the purpose of developing

proposals

4.1.2. Metadata implementers

4.1.3. UB itself

4.2. Requirements for proposals for "Recommended" and "Conforming" status

4.2.1. To be supplied by the proposers (see table below):

Proposal Requirements Table

Name	A suggested unique token for use in encodings
Label	A suggested human-readable label for the proposed term
Definition	The suggested definition of the term
Comment	Information concerning the possible application of the proposed term
Examples	Examples of use of the proposed term, making clear what type of literal values are expected
Type of term	Is the proposed term an "element," or an "element refinement" (as defined in http://dublincore.org/usage/documents/principles) [NOTE: Encoding schemes are registered using a separate process]
Term qualified	If the proposed term is an element refinement, which term does it qualify?
Why needed	A justification of the need for the proposed term
Working Group or community support	Demonstration and documentation that the proposed new term has substantial support of Working Group members as well as others in the relevant community. Evidence of such support can include votes held on mailing lists or in face-to-face meetings or positive endorsements from members of the DC-GENERAL mailing list.
Proposed status	Is the term proposed as Recommended or Conforming?
Related DCMI terms	A discussion of possible overlap with existing terms
Related non-DCMI terms	An annotated listing of related terms in non-DCMI metadata vocabularies
Impact on applications	An annotated listing of existing applications that could be affected by recognition of this term
About the proposers	A pointer to a description, in standard form (to be specified) of the working group or organization putting forward the proposal: its scope, aims, a brief history, current status, and a pointer to archives

4.2.2. To be supplied by the UB shepherd:
4.2.2.1. A summary history of the post-announcement discussion

4.3. Guidelines: The following criteria are offered as guidelines for developing a proposal -- they reflect criteria that the Usage Board will use in its decision-making. They do not constitute further requirements for the formal documentation of a proposal.

4.3.1. Among the major criteria used for evaluating a term proposal are the following:

4.3.1.1. Clarity

4.3.1.1.1. Can the term be clearly defined?

4.3.1.1.2. Can the semantics of the proposed element or element refinement be expressed precisely, unambiguously, and briefly?

4.3.1.2. Practicality

4.3.1.2.1. Is the term practical?
4.3.1.2.2. How difficult would it be for people creating metadata to comprehend the semantics of the proposed element or element refinement and to apply it reasonably in the description of resources?

4.3.1.3. Placement

4.3.1.3.1. Does the term refine an existing element?
4.3.1.3.2. If the proposed term is an element, can it reasonably be handled as effectively as an element refinement or encoding scheme for an existing element?
4.3.1.3.3. Are there alternative ways of implementing the term? Within the conceptual framework of the Dublin Core Element Set (i.e., element/element refinements and encoding schemes), are there alternative ways to achieve the ends sought?

4.3.1.4. Needs

4.3.1.4.1. Is there a clear requirement in existing implementations for the term in support of resource discovery?
4.3.1.4.2. Is there a demonstrated need for the proposed element or element refinement?
4.3.1.4.3. Are there existing implementations or encoding schemes, etc., which use the term?

4.3.1.5. Fits with other DCMI-maintained terms

4.3.1.5.1. Follows existing principles of refinement
4.3.1.5.2. Is well-formed
4.3.1.5.3. Does not conflict with or create ambiguity with regard to existing DCMI-maintained terms
4.3.1.5.4. Does not create problems for existing legacy implementations if those implementations have followed recommended practice

4.4. Decision tree for assessing the need for a new term

Decision Tree Table

Condition 1:	Can the need be solved with a vocabulary encoding scheme for an existing DCMI Element or Element Refinement?	If so, do that; else ·
Condition 2:	Can the need be solved through an application profile that references an element or element refinement from an existing and recognized non-DCMI namespace?	If so, do that; else ·
Condition 3:	Can the need be solved with a new refinement for an existing DCMI element?	If so, do that; else ·
Condition 4:	Create a new DCMI Element (and, if necessary, Element and Vocabulary Encoding Scheme) to meet the need.	

4.5. Process for Moving Proposals

4.5.1. Appointment of Shepherds

4.5.1.1. Each proposal shall be assigned a shepherd by the UB chair from among the UB membership.
4.5.1.2. Shepherds should have knowledge of the proposal issues or be connected to the WG originating the proposal.
4.5.1.3. Responsibilities

4.5.1.3.1. Monitor discussion on relevant lists (shepherds should be members of the relevant DC WG list during the time of consideration of a proposal and are encouraged to join in the discussion to ensure that all relevant issues are exposed during the discussion period).
4.5.1.3.2. Summarize the comment period discussion and points of contention of the proposal for the UB, either verbally at the meeting or in writing prior to the meeting (preferred).
4.5.1.3.3. Serve as liaison to the relevant WG or community during the time the proposal is under discussion and after a decision has been made.
4.5.1.3.4. Verify registration information for the DCMI Web Team.
4.5.1.3.5. Prepare draft of UB official decision on the proposal for review and approval by the UB.

4.5.2. Proposal is received by DCMI Managing Director or UB Chair.
4.5.3. Proposal is given preliminary review for completeness by DCMI Managing Director and UB Chair.
4.5.4. If complete and no revisions needed, proposal is circulated to UB members and announced for public comment by the Managing Director. A period of two weeks will be allowed between the date of the decision on completeness and the public announcement of the proposal to provide time for preparation of the supporting materials for public dissemination.
4.5.5. If incomplete or revisions needed, proposal is returned to originator, with request for revision or additional information.
4.5.6. Announcements

4.5.6.1. Announcements of comment period for proposals to be discussed by the UB shall be made in the following manner:

4.5.6.1.1. Announcement of the start of the public comment period shall be made on the DC General mailing list.
4.5.6.1.2. Comments regarding a proposal may be addressed to the relevant Working Group mailing list, the DC General mailing list or privately to the shepherd.

4.5.6.2. At the commencement of the public comment period, proposals for new terms must be moved to the DCMI Web site, given DCMI page headers and a status of 'Proposed term'.
4.5.6.3. Announcements of proposals shall be made by the shepherd.
4.5.6.4. Announcements will include:

4.5.6.4.1. Links to relevant information to be considered with the proposal
4.5.6.4.2. Relevant deadlines for comments
4.5.6.4.3. Addresses for comment submission
4.5.6.4.4. Information about UB meeting at which the proposal will be discussed, including place, time, and how to request an invitation to participate

4.5.6.4.5. Name and contact information for the assigned shepherd
4.5.6.4.6. The announcement should ask specifically for communications supporting the proposal in order to guage the level of community support.

4.5.7. Communication Responsibility Table

Communication Responsibility Table

What	Where	Who	Comment
Proposal draft posted	WG list, DC-General	WG Chair	
Proposal added to UB agenda	UB Website, UB list	UB Chair	
Proposal announced for public comment	DC-General	DCMI Managing Director	
Usage Board Outcome	DC-General	DCMI Managing Director	

4.5.8. Comment period

4.5.8.1. Comment period on proposals should be managed on the DC-General list.
4.5.8.2. Comment periods should be at least one month in length and commence at least six weeks before the UB meeting at which action is to be taken.
4.5.8.3. Public discussions of UB related issues during public comment periods should take place on DC-General or other working group mailing lists as specified in the announcement. The public discussion must start at least six weeks before the UB meeting at which the issues will be discussed.

4.5.9. Voting

4.5.9.1. Voting shall be limited to scheduled meetings and conference calls.
4.5.9.2. Voting shall be limited to UB members present at the meeting or conference call and able to participate in the discussion.
4.5.9.3. UB members who cannot be present may present their arguments for or against a proposal in writing prior to a meeting (this shall not constitute a vote).
4.5.9.4. UB members who cannot be present may explore other options with the chair, if they cannot be present for an important vote. In all cases, a vote may not be cast by a member who is not present, either physically or virtually, for the relevant discussion.
4.5.9.5. A proposal is approved if more than 50% of assigned votes in are in favor and fewer than 25% of assigned votes are against the proposal. Every effort will be made to achieve a firm consensus on a proposal before it is deemed approved.

4.6. Decisions of the UB are forwarded to the DCMI Directorate for endorsement and approval.
4.7. Registration of UB Decisions on Proposals

4.7.1. A document explaining the UB decision regarding a proposal will be written in a timely fashion by the shepherd and approved by the UB.

4.7.1.1. The decision will include brief statements of recommendations being issued and detailed explanations of UB decisions not to issue recommendations.
4.7.1.2. UB decisions will be in a form determined by the UB and numbered consecutively for the purpose of citation.
4.7.1.3. UB decisions must be sufficiently documented so that the

rationale for the decision is clear and useful in guiding the development of future proposals. This is particularly true where the decision rejects a proposal or recommends further action.
4.7.1.4. The DCMI Web Team will publish UB decisions in the Documents section of the DCMI Web site in a category named DCMI Usage Board Decisions.

4.7.2. Recommended terms will be put into the official DCMI documentation by the UB Chair.

5. Proposals for Registration of Encoding Schemes

5.1. Submissions of new encoding schemes will be received on the UB list via a Web form
5.2. UB members will "claim" responsibility to shepherd submissions based on:

5.2.1. Their knowledge of a particular scheme
5.2.2. Their knowledge of the language used in the scheme
5.2.3. Their interest or knowledge of a particular subject or topical area covered by the scheme
5.2.4. The time they have available for such tasks

5.3. Submissions unclaimed after one week will be assigned to a UB member by the chair.
5.4. The UB chair will not shepherd individual submissions, but will keep track of submissions and ensure that all are resolved in some manner.
5.5. The shepherd will be responsible for verifying the submitted information:

5.5.1. Name of the scheme
5.5.2. Availability and maintenance status
5.5.3. Appropriateness of the maintenance agency
5.5.4. Uniqueness and appropriateness of the proposed token
5.5.5. Possible use with elements not specified in the proposal

5.6. If necessary, the shepherd will initiate contact with the maintenance agency in the case of questions or concerns about the status of the scheme, the proposed token, or to clarify the submission.
5.7. The shepherd will edit the submission and complete the registration process by submitting the information to the DCMI Web Team.
5.8. The DCMI Web Team will report to the UB list when registration has been completed.
5.9. The UB chair will prepare a monthly report of all new schemes.
5.10. Communication Responsibility Table

Communication Responsibility Table

What	Where	Who	Comment
Scheme submission	UB List	Shepherd Submission Tool	
Scheme registration	UB List	Sheperd Submission Tool	Shepherd may announce to relevant list
Digest of scheme registrations	DC-General	UB-Chair	To be automated by DCMI Web Team

6. Proposals for Registration of Application Profiles

6.1. Sources of proposals

6.1.1. DCMI working groups

6.1.1.1. Existing working groups or working groups established for the purpose of developing proposals

6.1.2. Metadata implementers

6.1.2.1. Established projects or research groups

6.1.3. UB itself

6.2. For the purposes of review by the Usage Board:

6.2.1. The Usage Board is interested in reviewing application profiles that make substantial use of Dublin Core elements. The review of application profiles by the Usage Board serves to:

6.2.1.1. analyze the usage of Dublin Core within significant implementations;
6.2.1.2. assign a DCMI stamp of approval;
6.2.1.3. promote the sharing of application profiles between communities; and
6.2.1.4. identify new terms as candidates for inclusion in DCMI namespaces.

6.2.2. Application profiles must provide, for each term, an identifier of the element set where it is defined, ideally in the form of URIs for individual terms.
6.2.3. If the terms in an application profile describe anything other than generic "resources" (the typical domain of Dublin Core), the application profile must make this clear. This is particularly important if an application profile is based on a data model that describes multiple classes of resources, such as agents or collections.
6.2.4. It is recommended that application profiles be prepared using the Dublin Core Application Profile guidelines published by CEN [http://www.cenorm.be/cenorm/ businessdomains/ businessdomains/isss/cwa/cwa14855.asp].
6.2.5. Each application profile must provide, or point to, a short text that describes:

6.2.5.1. The context and purposes in which the application profile is used or is likely to be used.
6.2.5.2. The organizations or individuals involved in its development and a capsule history thereof.
6.2.5.3. Any arrangements, policies, or intentions regarding the future development and maintenance of the application profile.

6.3. Review of Application Profiles by the Usage Board

6.3.1. An application profile is "well-formed" if it is presented in accordance with the broad and flexible requirements outlined above. These presentation requirements may become more specific as "good practice" emerges over time.
6.3.2. Usage Board review focuses on the use of terms related to Dublin Core terms and on any data models that provide a context for those terms. The Usage Board is agnostic about the use of terms not directly related to Dublin Core; strictly speaking such terms are outside the scope of Usage Board review.
6.3.3. The use of terms related to Dublin Core (such as refinements of Dublin Core elements, or Dublin Core elements that have been constrained for particular contexts) will be evaluated from the standpoint of semantic conformance, grammatical principle (eg, "dumb-down"), clarity, and good practice.

6.4. Publication and use of Usage Board Reviews

6.4.1. An application profiles that "pass" review will be assigned the status of 'conforming'.

6.4.1.1. The status of 'conforming' indicates a Usage Board assessment of the application profile as of the date of its submission for review.
6.4.1.2. Changes to already 'conforming' application profiles require further Usage Board review of the application profile in whole or in part according to the processes and criteria outlined in sections 6.1through 6.3.

6.4.2. For application profiles that "pass" review, the Usage Board will publish a Review on a Web page for application profiles.
6.4.3. Each Review will include, at a minimum:

6.4.3.1. Any comments from the Usage Board on the application profile.
6.4.3.2. Pointers to locally archived copies of the application profile as originally submitted and (if necessary) as subsequently amended in light of Usage Board comments.
6.4.3.3. A pointer to the "latest version" of an application profile held by its maintainers.

6.5. Review represents a form of recognition, and its URL will be persistent for purposes of citation.

Part 3-Usage Board: Internal Processes

7. Changes to Usage Board Procedures

Metadata associated with this resource: http://dublincore.org/usage/documents/process/index.shtml.rdf

Using Dublin Core

Creator: Diane Hillmann

Date Issued: 2003-08-26
Identifier: http://dublincore.org/documents/2003/08/26/usageguide/
Replaces: http://dublincore.org/documents/2001/04/12/usageguide/
Is Replaced By: Not applicable
Latest Version: http://dublincore.org/documents/usageguide/
Translations: http://dublincore.org/resources/translations/
Status of Document: DCMI Recommended Resource

Description of Document: This document is intended as an entry point for users of Dublin Core. For non-specialists, it will assist them in creating simple descriptive records for information resources (for example, electronic documents). Specialists may find the document a useful point of reference to the documentation of Dublin Core, as it changes and grows.

Table of Contents

1. Introduction

1.1. What is Metadata?

Metadata has been with us since the first librarian made a list of the items on a shelf of handwritten scrolls. The

term "meta" comes from a Greek word that denotes "alongside, with, after, next." More recent Latin and English usage would employ "meta" to denote something transcendental, or beyond nature. Metadata, then, can be thought of as data about other data. It is the Internet-age term for information that librarians traditionally have put into catalogs, and it most commonly refers to descriptive information about Web resources.

A metadata record consists of a set of attributes, or elements, necessary to describe the resource in question. For example, a metadata system common in libraries -- the library catalog -- contains a set of metadata records with elements that describe a book or other library item: author, title, date of creation or publication, subject coverage, and the call number specifying location of the item on the shelf.

The linkage between a metadata record and the resource it describes may take one of two forms:

1. elements may be contained in a record separate from the item, as in the case of the library's catalog record; or
2. the metadata may be embedded in the resource itself.

Examples of embedded metadata that is carried along with the resource itself include the Cataloging In Publication (CIP) data printed on the verso of a book's title page; or the TEI header in an electronic text. Many metadata standards in use today, including the Dublin Core standard, do not prescribe either type of linkage, leaving the decision to each particular implementation.

Although the concept of metadata predates the Internet and the Web, worldwide interest in metadata standards and practices has exploded with the increase in electronic publishing and digital libraries, and the concomitant "information overload" resulting from vast quantities of undifferentiated digital data available online. Anyone who has attempted to find information online using one of today's popular Web search services has likely experienced the frustration of retrieving hundreds, if not thousands, of "hits" with limited ability to refine or make a more precise search. The wide scale adoption of descriptive standards and practices for electronic resources will improve retrieval of relevant resources in any venue where information retrieval is critical. As noted by Weibel and Lagoze, two leaders in the fields of metadata development and digital libraries:

> "The association of standardized descriptive metadata with networked objects has the potential for substantially improving resource discovery capabilities by enabling field-based (e.g., author, title) searches, permitting indexing of non-textual objects, and allowing access to the surrogate content that is distinct from access to the content of the resource itself." (Weibel and Lagoze, 1997)

In the last years we have also seen an increase in the application of Dublin Core metadata in more closed environments. There are implementations where Dublin Core metadata is used to describe resources held, owned or produced by companies, governments and international organisations to supporting portal services or internal knowledge management. There are also implementations where Dublin Core metadata is used as a common exchange format supporting the aggregation of collections of metadata, such as the case of the Open Archive Initiative. In these cases, like in the open environment of the Web, the concept of standardized descriptive metadata provides a powerful mechanism to improve retrieval for specific applications and specific user communities. It is this need for "standardized descriptive metadata" that the Dublin Core addresses.

1.2. What is the Dublin Core?

The Dublin Core metadata standard is a simple yet effective element set for describing a wide range of networked resources. The Dublin Core standard includes two levels: Simple and Qualified. Simple Dublin Core comprises fifteen elements; Qualified Dublin Core includes three additional elements (Audience, Provenance and RightsHolder), as well as a group of element refinements (also called qualifiers) that refine the semantics of the elements in ways that may be useful in resource discovery. The semantics of Dublin Core have been established by an international, cross-disciplinary group of professionals from librarianship, computer science, text encoding, the museum community, and other related fields of scholarship and practice.

Another way to look at Dublin Core is as a "small language for making a particular class of statements about resources". In this language, there are two classes of terms -- elements (nouns) and qualifiers (adjectives) -- which can be arranged into a simple pattern of statements. The resources themselves are the implied subjects in this language. (For additional discussion of Dublin Core Grammar, see "DCMI Grammatical Principles") In the diverse world of the Internet, Dublin Core can be seen as a "metadata pidgin for digital tourists": easily grasped, but not necessarily up to the task of expressing complex relationships or concepts.

The Dublin Core basic element set is outlined in Section 4. Each element is optional and may be repeated. Most elements also have a limited set of qualifiers or refinements, attributes that may be used to further refine (not

extend) the meaning of the element. The Dublin Core Metadata Initiative (DCMI) has established standard ways to refine elements and encourage the use of encoding and vocabulary schemes. The full set of elements and element refinements conforming to DCMI "best practice" is available, with a formal registry available as well.

Three other Dublin Core principles bear mentioning here, as they are critical to understanding how to think about the relationship of metadata to the underlying resources they describe.

1. The One-to-One Principle. In general Dublin Core metadata describes one manifestation or version of a resource, rather than assuming that manifestations stand in for one another. For instance, a jpeg image of the Mona Lisa has much in common with the original painting, but it is not the same as the painting. As such the digital image should be described as itself, most likely with the creator of the digital image included as a Creator or Contributor, rather than just the painter of the original Mona Lisa. The relationship between the metadata for the original and the reproduction is part of the metadata description, and assists the user in determining whether he or she needs to go to the Louvre for the original, or whether his/her need can be met by a reproduction.

2. The Dumb-down Principle. The qualification of Dublin Core properties is guided by a rule known colloquially as the Dumb-Down Principle. According to this rule, a client should be able to ignore any qualifier and use the value as if it were unqualified. While this may result in some loss of specificity, the remaining element value (minus the qualifier) must continue to be generally correct and useful for discovery. Qualification is therefore supposed only to refine, not extend the semantic scope of a property.

3. Appropriate values. Best practice for a particular element or qualifier may vary by context, but in general an implementor cannot predict that the interpreter of the metadata will always be a machine. This may impose certain constraints on how metadata is constructed, but the requirement of usefulness for discovery should be kept in mind.

Although the Dublin Core was originally developed with an eye to describing document-like objects (because traditional text resources are fairly well understood), DC metadata can be applied to other resources as well. Its suitability for use with particular non-document resources will depend to some extent on how closely their metadata resembles typical document metadata and also what purpose the metadata is intended to serve. (Implementors interested in using Dublin Core for diverse resources are encouraged to browse the Dublin Core Projects pages for ideas on using Dublin Core metadata for their resources.)

Dublin Core has as its goals:

Simplicity of creation and maintenance

> The Dublin Core element set has been kept as small and simple as possible to allow a non-specialist to create simple descriptive records for information resources easily and inexpensively, while providing for effective retrieval of those resources in the networked environment.

Commonly understood semantics

> Discovery of information across the vast commons of the Internet is hindered by differences in terminology and descriptive practices from one field of knowledge to the next. The Dublin Core can help the "digital tourist" -- a non-specialist searcher -- find his or her way by supporting a common set of elements, the semantics of which are universally understood and supported. For example, scientists concerned with locating articles by a particular author, and art scholars interested in works by a particular artist, can agree on the importance of a "creator" element. Such convergence on a common, if slightly more generic, element set increases the visibility and accessibility of all resources, both within a given discipline and beyond.

International scope

> The Dublin Core Element Set was originally developed in English, but versions

are being created in many other languages, including Finnish, Norwegian, Thai, Japanese, French, Portuguese, German, Greek, Indonesian, and Spanish. The DCMI Localization and Internationalization Special Interest Group is coordinating efforts to link these versions in a distributed registry.

Although the technical challenges of internationalization on the World Wide Web have not been directly addressed by the Dublin Core development community, the involvement of representatives from virtually every continent has ensured that the development of the standard considers the multilingual and multicultural nature of the electronic information universe.

Extensibility

While balancing the needs for simplicity in describing digital resources with the need for precise retrieval, Dublin Core developers have recognized the importance of providing a mechanism for extending the DC element set for additional resource discovery needs. It is expected that other communities of metadata experts will create and administer additional metadata sets, specialized to the needs of their communities. Metadata elements from these sets could be used in conjunction with Dublin Core metadata to meet the need for interoperabilbility. The DCMI Usage Board is presently working on a model for accomplishing this in the context of "application profiles."

Rachel Heery and Manjula Patel, in their article "Application profiles: mixing and matching metadata schemas" define an application profile as:

> " ... schemas which consist of data elements drawn from one or more namespaces, combined together by implementors, and optimised for a particular local application."

This model allows different communities to use the DC elements for core descriptive information, and allowing domain specific extensions which make sense within a more limited arena.

1.3. The Purpose and Scope of This Guide

This document is intended to be an entry point for users of Dublin Core. For non-specialists, it will assist in creating simple descriptive records for information resources (for example, electronic documents, JPEG images, video clips). Specialists may find the document a useful point of reference to the documentation of Dublin Core, as it changes and grows.

"Using Dublin Core" will show in a non-technical fashion how Dublin Core metadata may be used by anyone to make their material more accessible. It discusses the principles, structure and content of Dublin Core metadata elements, how to use them in composing a complete Dublin Core metadata record, as well as how to qualify elements to support use by a wide variety of communities.

Another important goal of this document is to promote "best practices" for describing resources using the Dublin Core element set. The Dublin Core community recognizes that consistency in creating metadata is an important key to achieving optimal retrieval and intelligible display across disparate sources of descriptive records. Inconsistent metadata effectively hides desired records, resulting in uneven, unpredictable or incomplete search results.

As a general introduction, this document is necessarily brief, and cannot address all the issues implementors may encounter while planning their use of metadata. Several avenues remain for those who have additional questions beyond those addressed in this guide.

1. 1. Appended to this guide are references to relevant articles and other resources, including those with more technical guidance for implementors
2. The Dublin Core Website contains references to additional documents and resources of the DCMI community and ways for implementors to become involved in the DCMI
3. Specific questions can be addressed to AskDCMI. In addition to fielding questions, the AskDCMI service maintains a searchable archive of already answered questions and links to additional resources.

2. Syntax Issues

The Dublin Core Abstract Modelprovides a reference model against which particular DC encoding guidelines can be compared, independent of any particular encoding syntax. Such a reference model allows implementors to gain a better understanding of the kinds of descriptions they are trying to encode and facilitates the development of better mappings and translations between different syntaxes. Although the document is primarily aimed at the developers of software applications that support Dublin Core metadata, anyone who is considering implementing Dublin Core -- particularly those contemplating extending DC in any way -- could usefully review the document. Those involved in developing new syntax encoding guidelines for Dublin Core metadata or developing metadata application profiles based on the Dublin Core should also become familiar with the DC Abstract Model.

In this guide, we have chosen to represent Dublin Core examples in a "generic" form (Element="value"). Examples of other syntaxes, including: HTML or XHTML (the Web's Hypertext Markup Language format), RDF/XML (the Resource Description Framework using eXtensable Markup Language) and in plain XML can be found in syntax-specific documents available on the DCMI Website. Some are also referenced within this document and in the Bibliography Section of this guide.

Syntax choices depend on a number of variables, and "one size fits all" prescriptions rarely apply. When considering an appropriate syntax, it is important to note that Dublin Core concepts and semantics are designed to be syntax independent, are equally applicable in a variety of contexts, as long as the metadata is in a form suitable for interpretation both by search engines and by human beings.

2.1. HTMLand XHTLM

HTML or XHTML can be used to express either simple or qualified Dublin Core, although there are limitations inherent in representing refinements in HTML. Specific instructions for expressing Dublin Core in HTML can be found in the following DCMI document:

1. Expressing Qualified Dublin Core in HTML/XHTML meta and link elements

2.2. RDF/XML

RDF (Resource Description Framework) allows multiple metadata schemes to be read by humans as well as parsed by machines. It uses XML (EXtensible Markup Language) to express structure thereby allowing metadata communities to define the actual semantics. This decentralized approach recognizes that no one scheme is appropriate for all situations, and further that schemes need a linking mechanism independent of a central authority to aid description, identification, understanding, usability, and/or exchange.

RDF allows multiple objects to be described without specifying the detail required. The underlying glue, XML, simply requires that all namespaces be defined and once defined, they can be used to the extent needed by the provider of the metadata.

For example:

```
<rdf:RDF
   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
   xmlns:dc="http://purl.org/dc/elements/1.1/">

   <rdf:Description rdf:about="http://media.example.com/audio/guide.ra">

      <dc:creator>Rose Bush</dc:creator>
      <dc:title>A Guide to Growing Roses</dc:title>
      <dc:description>Describes process for planting and nurturing different kinds of rose bushes.</dc:description>
      <dc:date>2001-01-20</dc:date>

   </rdf:Description>
</rdf:RDF>
```

This simple example uses Dublin Core by itself to describe an audio recording of a guide to growing rose bushes. With XML or RDF/XML, Dublin Core can potentially be mixed with other metadata vocabularies. For example, the simple Dublin Core description above might be used alongside other vocabularies such as vCard that can describe the author's affiliation and contact information, or a more specialized "rose description" vocabulary that described

the rose bushes in greater detail.

DCMI has made available several recommendations specifically about using these syntaxes:

1. Guidelines for Implementing Dublin Core in XML
2. Expressing Simple Dublin Core in RDF/XML
3. Expressing Qualified Dublin Core in RDF/XML (Proposed Recommendation)

2.3. Metadata Storage and Maintenance Issues

Some implementations using Dublin Core have chosen to embed their metadata within the resource itself. This approach is taken most often with documents encoded using HTML, but is also sometimes possible with other kinds of documents. Simple tools have been developed to make provision of Dublin Core metadata within HTML encoded pages fairly easy. One such tool, DC.dot, extracts metadata information from an HTML document, and formats it so that it can be edited, then cut and pasted back into the HTML header of the original document.

On the other hand, metadata can be stored in any kind of database, and provide a link to the described resource rather than be embedded within it. This approach is likely to be most practical for many non-textual resources, and is increasingly used for text as well, primarily to support easier maintenance and sharing of metadata.

Each of these approaches have their advantages and disadvantages, and the balance point changes as implementations become larger and more diverse, and also as the metadata ages over time.

3. Element Content and Controlled Vocabularies

Each Dublin Core element is optional and repeatable, and there is no defined order of elements. The ordering of multiple occurrences of the same element (e.g., Creator) may have a significance intended by the provider, but ordering is not guaranteed to be preserved in every user environment. Ordering or sequencing may be syntax dependent; for instance, RDF/XML supports ordering, but HTML does not.

Content data for some elements may be selected from a "controlled vocabulary," which is a limited set of consistently used and carefully defined terms. This can dramatically improve search results because computers are good at matching words character by character but weak at understanding the way people refer to one concept using different words, i.e. synonyms. Without basic terminology control, inconsistent or incorrect metadata can profoundly degrade the quality of search results. For example, without a controlled vocabulary, "candy" and "sweet" might be used to refer to the same concept. Controlled vocabularies may also reduce the likelihood of spelling errors when recording metadata.

One cost of a controlled vocabulary is the necessity for an administrative body to review, update and disseminate the vocabulary. For example, the US Library of Congress Subject Headings (LCSH) and the US National Library of Medicine Medical Subject Headings (MeSH) are formal vocabularies, indispensable for searching rigorously cataloged collections. However, both require significant support organizations. Another cost is having to train searchers and creators of metadata so that they know when using MeSH, for example, to enter "myocardial infarction" instead of the more colloquial "heart attack." More sophisticated implementations can make such tasks much easier, but the controlled vocabulary terms must be available for them to apply.

Using controlled vocabularies can be done most effectively using encoding schemes. Without an encoding scheme specifically designated, a subject which might very well be carefully selected from a particular controlled vocabulary cannot be distinguished from a simple keyword.

4. The Elements

5. Dublin Core Qualifiers

6. Glossary

7. Bibliography

Metadata associated with this resource: http://dublincore.org/documents/usageguide/index.shtml.rdf

Using Dublin Core - The Elements

Creator:	Diane Hillmann
Date Issued:	2003-08-26
Identifier:	http://dublincore.org/documents/2003/08/26/usageguide/elements.shtml
Replaces:	http://dublincore.org/documents/2001/04/12/usageguide/sectc.shtml
Is Replaced By:	Not applicable
Is Part Of:	http://dublincore.org/documents/2003/08/26/usageguide/
Latest Version:	http://dublincore.org/documents/usageguide/
Status of Document:	DCMI Recommended Resource
Description of Document:	This document is intended as an entry point for users of Dublin Core. For non-specialists, it will assist them in creating simple descriptive records for information resources (for example, electronic documents). Specialists may find the document a useful point of reference to the documentation of Dublin Core, as it changes and grows.

4. The Elements

This section lists each element by its full name and label. For each element there are guidelines to assist in creating metadata content, whether it is done "from scratch" or by converting an existing record in another format.

In the element descriptions below, a formal single-word label is specified to make the syntactic specification of elements simpler for encoding schemes. Although some environments, such as HTML, are not case-sensitive, it is recommended best practice always to adhere to the case conventions in the element names given below to avoid conflicts in the event that the metadata is subsequently converted to a case-sensitive environment, such as XML.

Some information may appear to belong in more than one metadata element. While there will normally be a clear preferred choice, there is potential semantic overlap between some elements. Consequently, there will occasionally be some judgment required from the person assigning the metadata.

4.1. Title

Label: Title

Element Description: The name given to the resource. Typically, a Title will be a name by which the resource is formally known.

Guidelines for creation of content:

If in doubt about what constitutes the title, repeat the Title element and include the variants in second and subsequent Title iterations. If the item is in HTML, view the source document and make sure that the title identified in the title header (if any) is also included as a Title.

Examples:

Title="A Pilot's Guide to Aircraft Insurance"
Title="The Sound of Music"
Title="Green on Greens"
Title="AOPA's Tips on Buying Used Aircraft"

4.2. Subject

Label: Subject and Keywords

Element Description: The topic of the content of the resource. Typically, a Subject will be expressed as keywords or key phrases or classification codes that describe the topic of the resource. Recommended best practice is to select a value from a controlled vocabulary or formal classification scheme.

Guidelines for creation of content:

Select subject keywords from the Title or Description information, or from within a text resource. If the subject of the item is a person or an organization, use the same form of the name as you would if the person or organization were a Creator or Contributor.

In general, choose the most significant and unique words for keywords, avoiding those too general to describe a particular item. Subject might include classification data if it is available (for example, Library of Congress Classification Numbers or Dewey Decimal numbers) or controlled vocabularies (such as Medical Subject Headings or Art and Architecture Thesaurus descriptors) as well as keywords.

When including terms from multiple vocabularies, use separate element iterations. If multiple vocabulary terms or keywords are used, either separate terms with semi-colons or use separate iterations of the Subject element.

Examples:

Subject="Aircraft leasing and renting"
Subject="Dogs"
Subject="Olympic skiing"
Subject="Street, Picabo"

4.3. Description

Label: Description

Element Description: An account of the content of the resource. Description may include but is not limited to: an abstract, table of contents, reference to a graphical representation of content or a free-text account of the content.

Guidelines for creation of content:

Since the Description field is a potentially rich source of indexable terms, care should be taken to provide this element when possible. Best practice recommendation for this element is to use full sentences, as description is often used to present information to users to assist in their selection of appropriate resources from a set of search results.

Descriptive information can be copied or automatically extracted from the item if there is no abstract or other structured description available. Although the source of the description may be a web page or other structured text with presentation tags, it is generally not good practice to include HTML or other structural tags within the Description element. Applications vary considerably in their ability to interpret such tags, and their inclusion may negatively affect the interoperability of the metadata.

Examples:

Description="Illustrated guide to airport markings and lighting signals, with particular reference to SMGCS (Surface Movement Guidance and Control System) for airports with low visibility conditions."

Description="Teachers Domain is a multimedia library for K-12 science educators, developed by WGBH through funding from the National Science Foundation as part of its National Science Digital Library initiative. The site offers a wealth of classroom-ready instructional resources, as well as online professional development materials and a set of tools which allows teachers to manage, annotate, and share the materials they use in classroom teaching."

4.4. Type

Label: Resource Type

Element Description: The nature or genre of the content of the resource. Type includes terms describing general categories, functions, genres, or aggregation levels for content. Recommended best practice is to select a value from a controlled vocabulary (for example, the DCMIType vocabulary). To describe the physical or digital manifestation of the resource, use the FORMAT element.

Guidelines for content creation:

If the resource is composed of multiple mixed types then multiple or repeated Type elements should be used to describe the main components.

Because different communities or domains are expected to use a variety of type vocabularies, best practice to ensure interoperability is to include at least one general type term from the DCMIType vocabulary in addition to the domain specific type term(s), in separate Type element iterations.

Examples:

Type="Image"
Type="Sound"
Type="Text"
Type="simulation"

Note: The first three values are taken from the DCMI Type Vocabulary, and follow the capitalization conventions for that vocabulary. The last value is a term from an unspecified source.

The item described is an *Electronic art exhibition catalog:*

Type="Image"
Type="Text"
Type="Exhibition catalog"

Note: The first two values are taken from the DCMI Type Vocabulary, and follow the capitalization conventions for that vocabulary. The last value is a term from an unspecified source.

The item described is a *Multimedia educational program with interactive assignments:*

Type="Image"
Type="Text"
Type="Software"
Type="Interactive Resource"

Note: All values in this example are taken from the DCMI Type Vocabulary, and follow the capitalization conventions for that vocabulary.

4.5. Source

Label: Source

Element Description: A Reference to a resource from which the present resource is derived. The present resource may be derived from the Source resource in whole or part. Recommended best practice is to reference the resource by means of a string or number conforming to a formal identification system.

Guidelines for content creation:

In general, include in this area information about a resource that is related intellectually to the described resource but does not fit easily into a Relation element.

Examples:

> Source="RC607.A26W574 1996" [where "RC607.A26W574 1996" is the call number of the print version of the resource, from which the present version was scanned]
>
> Source="Image from page 54 of the 1922 edition of Romeo and Juliet"

4.6. Relation

Label: Relation

Element Description: A reference to a related resource. Recommended best practice is to reference the resource by means of a string or number conforming to a formal identification system.

Guidelines for content creation:

Relationships may be expressed reciprocally (if the resources on both ends of the relationship are being described) or in one direction only, even when there is a refinement available to allow reciprocity. If text strings are used instead of identifying numbers, the reference should be appropriately specific. For instance, a formal bibliographic citation might be used to point users to a particular resource.

Because the refined terms used with Relation provide significantly more information to a user than the unqualified use of Relation, implementers who are describing heavily interrelated resources might choose to use qualified Dublin Core.

Examples:

> Title="Reading Turgenev"
> Relation="Two Lives" [Resource is a collection of two novellas, one of which is "Reading Turgenev"]
> *[Relationship described is **IsPartOf**.*

[Part/Whole relations are those in which one resource is a physical or logical part of another]

> Title="Candle in the Wind"
> Subject="Diana, Princess of Wales"
> Date="1997"
> Creator="John, Elton"
> Type="sound"
> Description="Tribute to a dead princess."
> Relation="Elton John's 1976 song Candle in the Wind"
> *[Relationship described is **IsVersionOf**.*

[Version relations are those in which one resource is an historical state or edition, of another resource by the same creator]

> Title="Electronic AACR2"
> Relation="Anglo-American Cataloging Rules, 2nd edition"

*[Relationship described is **IsFormatOf**]*

Title="Landsat TM dataset of Arnhemland, NT, Australia"
Relation="arnhem.gif"
[Relationship described is **HasFormat**]

[Format transformation relations are those in which one resource has been derived from another by a reproduction or reformatting technology which is not fundamentally an interpretation but intended to be a representation.]

Title="Morgan's Ancient Society"
Relation="Engels' Origin of the Family, Private Property and the State"
*[Relationship described is **IsReferencedBy**]*

Title="Nymphet Mania"
Relation="References Adrian Lyne's 'Lolita'"
*[Relationship described is **References**]*

[Reference relations are those in which the author of one resource cites, acknowledges, disputes or otherwise make claims about another resource.]

Title="Peter Carey's novel Oscar and Lucinda"
Relation="1998 movie Oscar and Lucinda"
*[Relationship described is **IsBasisFor**]*

Title="The movie My Fair Lady"
Relation="Shaw's play Pygmalion"
*[Relationship described is **IsBasedOn**]*

[Creative relations are those in which one resource is a performance, production, derivation, adaptation or interpretation of another resource.]

Title="Dead Ringer"
Relation="Gemstar e-book"
*[Relationship described is **Requires**]*

[Dependency relations are those in which one resource requires another resource for its functioning, delivery, or content and cannot be used without the related resource being present.]

4.7. Coverage

Label: Coverage

Element Description: The extent or scope of the content of the resource. Coverage will typically include spatial location (a place name or geographic co-ordinates), temporal period (a period label, date, or date range) or jurisdiction (such as a named administrative entity). Recommended best practice is to select a value from a controlled vocabulary (for example, the Thesaurus of Geographic Names [Getty Thesaurus of Geographic Names, http://www.getty.edu/research/tools/vocabulary/tgn/]). Where appropriate, named places or time periods should be used in preference to numeric identifiers such as sets of co-ordinates or date ranges.

Guidelines for content creation:

Whether this element is used for spatial or temporal information, care should be taken to provide consistent information that can be interpreted by human users, particularly in order to provide interoperability in situations where sophisticated geographic or time-specific searching is not supported. For most simple applications, place names or coverage dates might be most useful. For more complex applications, consideration should be given to using an encoding scheme that supports appropriate specification of information, such as DCMI Period, DCMI Box or DCMI Point.

Examples:

Coverage="1995-1996"
Coverage="Boston, MA"
Coverage="17th century"
Coverage="Upstate New York"

4.8. Creator

Label: Creator

Element Description: An entity primarily responsible for making the content of the resource. Examples of a Creator include a person, an organization, or a service. Typically the name of the Creator should be used to indicate the entity.

Guidelines for creation of content:

Creators should be listed separately, preferably in the same order that they appear in the publication. Personal names should be listed surname or family name first, followed by forename or given name. When in doubt, give the name as it appears, and do not invert.

In the case of organizations where there is clearly a hierarchy present, list the parts of the hierarchy from largest to smallest, separated by full stops and a space. If it is not clear whether there is a hierarchy present, or unclear which is the larger or smaller portion of the body, give the name as it appears in the item.

If the Creator and Publisher are the same, do not repeat the name in the Publisher area. If the nature of the responsibility is ambiguous, the recommended practice is to use Publisher for organizations, and Creator for individuals. In cases of lesser or ambiguous responsibility, other than creation, use Contributor.

Examples:

Creator="Shakespeare, William"
Creator="Wen Lee"
Creator="Hubble Telescope"
Creator="Internal Revenue Service. Customer Complaints Unit"

4.9. Publisher

Label: Publisher

Element Description: The entity responsible for making the resource available. Examples of a Publisher include a person, an organization, or a service. Typically, the name of a Publisher should be used to indicate the entity.

Guidelines for content creation:

The intent of specifying this field is to identify the entity that provides access to the resource. If the Creator and Publisher are the same, do not repeat the name in the Publisher area. If the nature of the responsibility is ambiguous, the recommended practice is to use Publisher for organizations, and Creator for individuals. In cases of ambiguous responsibility, use Contributor.

Examples:

Publisher="University of South Where"
Publisher="Funky Websites, Inc."
Publisher="Carmen Miranda"

4.10. Contributor

Label: Contributor

Element Description: An entity responsible for making contributions to the content of the resource. Examples of a Contributor include a person, an organization or a service. Typically, the name of a Contributor should be used to indicate the entity.

Guideline for content creation:

The same general guidelines for using names of persons or organizations as Creators apply here. Contributor is the most general of the elements used for "agents" responsible for the resource, so should be used when primary responsibility is unknown or irrelevant.

4.11. Rights

Label: Rights Management

Element Description: Information about rights held in and over the resource. Typically a Rights element will contain a rights management statement for the resource, or reference a service providing such information. Rights information often encompasses Intellectual Property Rights (IPR), Copyright, and various Property Rights. If the rights element is absent, no assumptions can be made about the status of these and other rights with respect to the resource.

Guidelines for content creation:

The Rights element may be used for either a textual statement or a URL pointing to a rights statement, or a combination, when a brief statement and a more lengthy one are available.

Examples:

Rights="Access limited to members"
Rights="http://cs-tr.cs.cornell.edu/Dienst/Repository/2.0/Terms"

4.12. Date

Label: Date

Element Description: A date associated with an event in the life cycle of the resource. Typically, Date will be associated with the creation or availability of the resource. Recommended best practice for encoding the date value is defined in a profile of ISO 8601 [Date and Time Formats, W3C Note, http://www.w3.org/TR/NOTE-datetime] and follows the YYYY-MM-DD format.

Guidelines for content creation:

If the full date is unknown, month and year (YYYY-MM) or just year (YYYY) may be used. Many other schemes are possible, but if used, they may not be easily interpreted by users or software.

Examples:

Date="1998-02-16"
Date="1998-02"
Date="1998"

4.13. Format

Label: Format

Element Description: The physical or digital manifestation of the resource. Typically, Format may include the media-type or dimensions of the resource. Examples of dimensions include size and duration. Format may be used to determine the software, hardware or other equipment needed to display or operate the resource.

Recommended best practice is to select a value from a controlled vocabulary (for example, the list of Internet Media Types [http://www.iana.org/assignments/media-types/] defining computer media formats).

Guidelines for content creation:

In addition to the specific physical or electronic media format, information concerning the size of a resource may be included in the content of the Format element if available. In resource discovery size, extent or medium of the resource might be used as a criterion to select resources of interest, since a user may need to evaluate whether they can make use of the resource within the infrastructure available to them.

When more than one category of format information is included in a single record, they should go in separate iterations of the element.

Examples:

Title="Dublin Core icon"
Identifier="http://purl.org/metadata/dublin_core/images/dc2.gif"
Type="Image"
Format="image/gif"
Format="4 kB"

Subject="Saturn"
Type="Image"
Format="image/gif 6"
Format="40 x 512 pixels"
Identifier="http://www.not.iac.es/newwww/photos/images/satnot.gif"

Title="The Bronco Buster"
Creator="Frederic Remington"
Type="Physical object"
Format="bronze"
Format="22 in."

4.14. Identifier

Label: Resource Identifier

Element Description: An unambiguous reference to the resource within a given context. Recommended best practice is to identify the resource by means of a string or number conforming to a formal identification system. Examples of formal identification systems include the Uniform Resource Identifier (URI) (including the Uniform Resource Locator (URL), the Digital Object Identifier (DOI) and the International Standard Book Number (ISBN).

Guidelines for content creation:

This element can also be used for local identifiers (e.g. ID numbers or call numbers) assigned by the Creator of the resource to apply to a particular item. It should not be used for identification of the metadata record itself.

Examples:

Identifier="http://purl.oclc.org/metadata/dublin_core/"
Identifier="ISBN:0385424728"
Identifier="H-A-X 5690B" [publisher number]

4.15. Language

Label: Language

Element Description: A language of the intellectual content of the resource. Recommended best practice for the values of the Languge element is defined by RFC 3066 [RFC 3066, http://www.ietf.org/rfc/rfc3066.txt] which, in conjunction with ISO 639 [ISO 639, http://www.oasis-open.org/cover/iso639a.html]), defines two- and three-letter primary language tags with optional subtags. Examples include "en" or "eng" for English, "akk" for Akkadian, and "en-GB" for English used in the United Kingdom.

Guidelines for content creation:

Either a coded value or text string can be represented here. If the content is in more than one language, the element may be repeated.

Examples:

Language="en"
Language="fr"
Language="Primarily English, with some abstracts also in French."
Language="en-US"

NOTE: Audience, Provenance and RightsHolder are elements, but not part of the Simple Dublin Core fifteen elements. Use Audience, Provenance and RightsHolder only when using Qualified Dublin Core.

4.16. Audience

Label: Audience

Element Description: A class of entity for whom the resource is intended or useful. A class of entity may be determined by the creator or the publisher or by a third party.

Guidelines for content creation:

Audience terms are best utilized in the context of formal or informal controlled vocabularies. None are presently recommended or registered by DCMI, but several communities of interest are engaged in setting up audience vocabularies. In the absence of recommended controlled vocabularies, implementors are encouraged to develop local lists of values, and to use them consistently.

Examples:

Audience="elementary school students"
Audience="ESL teachers"
Audience="deaf adults"

4.17. Provenance

Label: Provenance

Element Description: A statement of any changes in ownership and custody of the resource since its creation that are significant for its authenticity, integrity and interpretation. The statement may include a description of any changes successive custodians made to the resource.

Guidelines for content creation:

Examples:

Provenance="This copy once owned by Benjamin Spock."
Provenance="Estate of Hunter Thompson."
Provenance="Stolen in 1999; recovered by the Museum in 2003."

4.18. RightsHolder

Label: Rights Holder

Element Description: A person or organization owning or managing rights over the resource. Recommended best practice is to use the URI or name of the Rights Holder to indicate the entity.

Guidelines for content creation:

Since, for the most part, people are not currently referenced by URI, a person holding rights over a resource would be named using a text string. Organizations sometimes have websites, but URLs for these are not generally appropriate for use in this context, since they are not clearly identifying the organization, but rather the location of a website about the organization. Appropriate choices, therefore are a text string or a URI from an authority file that exposes URIs for organizational entities, and the best choice is whichever seems most appropriate in the context of the needs of the target user.

Examples:

RightsHolder="Stuart Weibel"
RightsHolder="info:lccn/n78089035"
RightsHolder="University of Bath"

Using Dublin Core - Dublin Core Qualifiers

Date Issued:	2003-08-26
Identifier:	http://dublincore.org/documents/2003/08/26/usageguide/qualifiers.shtml
Replaces:	Not Applicable
Is Replaced By:	Not Applicable
Latest version:	http://dublincore.org/documents/usageguide/
Translations:	http://dublincore.org/resources/translations/
Status of document:	DCMI Recommended Resource
Description of document:	This document describes the principles governing Dublin Core qualifiers, the two categories of qualifiers, and lists instances of qualifiers approved by the Dublin Core Usage Board as well as guidance for their use.

5. Dublin Core Qualifiers

This document presents in part the results of an ongoing process to develop exemplary terms extending or refining the original 15 elements of the Dublin Core Metadata Element Set (DCMES). The terms or "qualifiers" listed here were identified, generally in working groups of the Dublin Core Metadata Initiative, (DCMI) and judged by the DCMI Usage Board to be in conformance with principles of good practice for the qualification of Dublin Core metadata elements.

In determining the makeup of these qualifiers, preference was given to vocabularies, notations, and terms already maintained by established agencies. It should be emphasized that the list of externally-maintained vocabularies identified here is a preliminary list. There are many more controlled vocabularies or classification systems that are not on this list. Detail on currently recommended schemes are listed at: DCMI Encoding Schemes - a current list

.

Inevitably, there will be situations where an agent or client will encounter DCMES descriptions that use unfamiliar qualifiers developed by implementors for specialized local or domain-specific needs. The useful interpretation of such a DCMES description will depend on the ability of an application to ignore the unknown qualifiers and fall back on the broader meaning of the element in its unqualified form. The guiding principle for the qualification of Dublin Core elements, colloquially known as the "Dumb-Down Principle," is that a client should be able to ignore any qualifier and use the information as if it were unqualified. While this may result in some loss of specificity, the remaining element value (without the qualifier) should continue to be generally correct and useful for discovery.

It is expected that implementors will develop additional qualifiers for use within local applications or specific domains. Such qualifiers may not be understood by other applications. However, qualifiers that conform to the principles of qualification defined here are more likely to be reusable by other communities within the broader context of cross-domain discovery.

At the time of the ratification of this document, the DCMI recognizes two broad classes of qualifiers:

- **Element Refinement.** These qualifiers make the meaning of an element narrower or more specific. A refined element shares the meaning of the unqualified element, but with a more restricted scope. A client that does not understand a specific element refinement term should be able to ignore the qualifier and treat the metadata value as if it were an unqualified (broader) element.

The definitions of element refinement terms for qualifiers must be publicly available.

- **Encoding Scheme.** These qualifiers identify schemes that aid in the interpretation of an element value. These schemes include controlled vocabularies and formal notations or parsing rules. A value expressed using an encoding scheme will thus be a token selected from a controlled vocabulary (e.g., a term from a classification system or set of subject headings) or a string formatted in accordance with a formal notation (e.g., "2000-01-01" as the standard expression of a date). If an encoding scheme is not understood by a client or agent, the value may still be useful to a human reader. The definitive description of an encoding scheme for qualifiers must be clearly identified and available for public use.

All of the qualifiers listed in this document fall into one of these two categories. Specific guidance is given below for element refinements. If a particular encoding scheme is available for the element and or/element refinement, its application is generally described either in this document or in documentation available with the encoding scheme itself. Audience, Provenance and RightsHolder, which are at the element level but not one of the original 15 elements, are described along with the other elements.

Additional qualifier categories may evolve over time and with implementation experience. The qualifiers listed here do not constitute a closed set, designed to meet all of the descriptive needs of implementors. Rather, they form the foundation for a larger body of qualifiers that will evolve as additional qualifiers are developed by various communities, some of which may eventually be submitted to the DCMI Usage Board for review and approval. Implementors may deploy the qualifiers on these lists with confidence that they conform to the Dumb-Down Principle, and are encouraged to use these qualifiers as examples to guide development of local qualifiers for Dublin Core metadata elements.

Summary Refinement and Scheme Table

This summary of the element refinements and schemes is provided for the convenience of users. Terms in this summary may have the status of "recommended" or "conforming." The reference definitions and status indications may be found in DCMI Terms. Click on the term to go directly to the reference definition for that term.

DCMES Element	Element Refinement(s)	Element Encoding Scheme(s)
Title	Alternative	-
Creator	-	-
Subject	-	LCSH MeSH DDC LCC UDC
Description	Table Of Contents Abstract	-
Publisher	-	-
Contributor	-	-
Date	Created Valid Available Issued Modified Date Accepted Date Copyrighted Date Submitted	DCMI Period W3C-DTF
Type	-	DCMI Type Vocabulary
	-	IMT
Format	Extent	-
	Medium	-
Identifier	-	URI
	Bibliographic Citation	-
Source	-	URI
Language	-	ISO 639-2RFC 3066

Relation	Is Version Of Has Version Is Replaced By Replaces Is Required By Requires Is Part Of Has Part Is Referenced By References Is Format Of Has Format Conforms To	URI
Coverage	Spatial	DCMI Point ISO 3166 DCMI Box TGN
	Temporal	DCMI Period W3C-DTF
Rights	Access Rights	-
	License	URI
Audience	Mediator Education Level	-
Provenance	-	-
Rights Holder	-	-

Properties of Dublin Core Qualifiers

Dublin Core qualifiers have the following properties:

- **Name:** The unique token assigned to the qualifier.
- **Label:** The human-readable label assigned to the qualifier.
- **Definition:** A statement that represents the concept and essential nature of the qualifier.
- **Comment:** Additional information associated with the qualifier (if available).
- **See Also:** A link to more information about the qualifier (if available).

For the up-to-date specification of all metadata terms maintained by the Dublin Core Metadata Initiative, including elements, element refinements, encoding schemes, and vocabulary terms (the DCMI Type Vocabulary), see http://dublincore.org/documents/dcmi-terms/. In the listing below, the Name and Label attributes are the same as in the specification, but the Definition and Comment appear together as "Term Description", and guidance and examples are added.

Multiple Language Encodings of Dublin Core Entities

Dublin Core qualifiers will be expressed in languages other than English. A single invariant token assigned to each qualifier -- the Name property -- stands for a given qualifier concept irrespective of the language in which it is defined. This token can be incorporated into a URI to form a unique identifier for the qualifier. All other properties of a qualifier (Label, Definition, Comment, and aspects of See Also as appropriate) can be translated from English into any other language.

All other properties of Dublin Core entities (Label, Definition, Comment, and aspects of See Also as appropriate) will be expressed in the language and character set of the translation.

Element Refinements

These element refinement terms are extensions to the "Simple Dublin Core" 15 elements or to the additional element terms Audience, Provenance and RightsHolder.

Refinement(s) for element: Title

Alternative

Label: Alternative

Term description: Any form of the title used as a substitute or alternative to the formal title of the resource. This qualifier can include Title abbreviations as well as translations.

Guidelines for creation of content:

An alternative title can be used to provide access to secondary titles, but should only be used when a value is present in the Title element.

Examples:

> Alternative="AMA newsletter" (Title="American Meteorological Association newsletter")
> Alternative="Ocho semanas" (Title="Eight weeks")

Refinement(s) for element: Description

tableOfContents

Label: Table of Contents

Term description: A list of subunits of the content of the resource.

Guidelines for creation of content:

When a description of a resource consists of a list of the contents, whether from a menu or other mechanism, tableOfContents can be used to differentiate this list from descriptive text that is written in sentence form. This allows more options for display and indexing.

Examples:

> tableOfContents="Introduction; Vertebrates; Invertebrates; Molluscs"

Abstract

Label: Abstract

Term description: A summary of the content of the resource.

Guidelines for creation of content:

Used when a description of a resource consists of a formal abstract. For implementations where formal abstracts are preferred, using the specific term allows the label to better reflect the level of the description.

Examples:

> Abstract="This article describes the work of the IFB Chaos Committee, including a summary of its major findings."

Refinement(s) for element: Date

Date refinements are generally useful in situations where more than one date is needed, and the difference between the dates may be important to users. Note that the first five Date refinement terms were among the earlier ones approved by DCMI, and the naming convention of the time was not to include "date" as part of the refined term. The most recent ones reflect changes in the naming convention used, in which the name of the refined term expresses more clearly the relationship to the parent element. When using date refinements it can be unwise to insert a text string that repeats the distinction created by the refinement itself. For instance, the string "Valid 20010211" in a statement where the refinement "valid" is used might show up in a labelled display as: VALID: Valid 20010211.

Created

Label: Created

Term description: Date of creation of the resource.

Guidelines for creation of content:

If the date of creation of the resource is known, and that date is important to note specifically (e.g., there are other relevant dates to record), use the term Created for the creation date of the resource. Note that the "one-to-one" rule requires that the creation date be that of the resource being described, not any early version from which the current resource is derived.

Valid

Label: Valid

Term description: Date (often a range) of validity of a resource.

Guidelines for creation of content:

If the resource is only valid or relevant for a particular date or range of dates, the term Valid may be used to express those dates. This may be particularly important if the resource will be retained over time but its use is valid only during a particular period or until a particular date.

Available

Label: Available

Term description: Date (often a range) that the resource will become or did become available.

Guidelines for creation of content:

In general, the term Available should be used in the case of a resource for which the date of availability may be distinct from the date of creation, and the date of availability is relevant to the use of the resource.

Issued

Label: Issued

Term description: Date of formal issuance (e.g., publication) of the resource.

Guidelines for creation of content:

The term Issued should be applied when a formal date of issuance or publication is relevant to the resource, and is distinct from other dates that may be used with the resource.

Modified

Label: Modified

Term description: Date on which the resource was changed.

Guidelines for creation of content:

Modified dates may be used to record either all instances of modification or only the latest. When only one modified date is recorded, it is assumed to be the latest.

dateAccepted

Label: Date Accepted

Term description: Date of acceptance of the resource (e.g. of thesis by university department, of article by journal, etc.).

Guidelines for creation of content:

If, in the lifecycle of a resource, the date of acceptance by a formal body or entity is relevant to the use of the resource, dateAccepted may be used.

dateCopyrighted

Label: Date Copyrighted

Term description: Date of a statement of copyright.

Guidelines for creation of content:

If, in the lifecycle of a resource, the date of copyright is relevant to the use of the resource, dateCopyrighted may be used.

dateSubmitted

Label: Date Submitted

Term description: Date of submission of the resource (e.g. thesis, articles, etc.).

Guidelines for creation of content:

If, in the lifecycle of a resource, the date of submission to a body or entity is relevant to the use of the resource, dateSubmitted may be used.

Refinement(s) for element: Format

Extent

Label: Extent

Term description: The size or duration of the resource.

Guidelines for creation of content:

Because the refinement Extent is used in a variety of situations, it generally consists of both a numeric value and a caption that is needed to interpret the numeric value. Best practice is to separate the numeric value and the caption with a space, whether the caption appears before or after the value.

Examples:

Extent="folio"
Extent="899 Kb"
Extent="21 minutes"

Medium

Label: Medium

Term description: The material or physical carrier of the resource.

Guidelines for creation of content:

Medium is generally used when the resource is of a physical nature, for instance a painting or model, where the physical carrier or material used is relevant to the user. For instance, if the resource is a movie on DVD, and is only available as a physical object, it should be described as such. If it is available digitally, for download or presentation on a website, its format would be reflected in the Format element. Note that, because of the physical nature of materials described with this refinement, the encoding scheme "IMT" is not valid for use with Medium.

Examples:

Medium="cotton fabric with sequins"
Medium="bronze on wooden pedestal"
Medium="oil on wood"

Refinement(s) for element: Relation

Most of the refinements of Relation are expressed as "reciprocals" and may be used to link resources in two directions, though this is not required. Implementors need not describe both or all resources involved in a reciprocal relationship to express the relationship--in other words, they may describe a later version and relate it to the earlier without having the need or opportunity to describe the earlier, and vice versa. In some of the relationships below, maintaining reciprocality is more important. In others, one direction of the relationship is more relevant that the other. These differences will be mentioned in the guidelines for specific terms.

In All cases, either a string or a URI may be used as a value. If a URI is used, the scheme should be designated.

Examples for Relation refinements can be found with the Relation element. When using Relation refinements, do not use embedded text labels, as the examples illustrate.

isVersionOf

Label: Is Version Of

Term description: The described resource is a version, edition, or adaptation of the referenced resource. Changes in version imply substantive changes in content rather than differences in format.

Guidelines for creation of content:

Use only in cases where the relationship expressed is at the content level. Relationships need not be close for the relationship to be relevant. "West Side Story" is a version of "Romeo and Juliet" and that may be important enough in the context of the resource description to be expressed using isVersionOf. The Broadway Show and the movie of "West Side Story" also relate at a similar level, but the video and DVD of the movie are more usefully expressed at the level of format, the content being essentially the same.

See also isFormatOf.

hasVersion

Label: Has Version

Term description: The described resource is a version, edition, or adaptation of the referenced resource. Changes in version imply substantive changes in content rather than differences in format.

Guidelines for creation of content:

See isVersionOf for basic guidelines.

isReplacedBy

Label: Is Replaced By

Term description: The described resource is supplanted, displaced, or superseded by the referenced resource.

Guidelines for creation of content:

When establishing a chain of versions, where only one version is valid, the use of isReplacedBy and Replaces allows the relationship to be expressed and the user directed to the appropriate version. In this case, the reciprocal relationships are quite important.

Replaces

Label: Replaces

Term description: The described resource supplants, displaces, or supersedes the referenced resource.

Guidelines for creation of content:

See isReplacedBy for basic guidelines.

isRequiredBy

Label: Is Required By

Term description: The described resource is required by the referenced resource, either physically or logically.

Guidelines for creation of content:

In the case of IsRequiredBy and Requires, there is a clearer need to express the Requires relationship than the IsRequiredBy, though both can be useful. This relationship is most often seen in relationships between software and documents or applications and hardware and/or

software requirements.

Requires

Label: Requires

Term description: The described resource requires the referenced resource to support its function, delivery, or coherence of content.

Guidelines for creation of content:

See isRequiredBy for basic guidelines.

isPartOf

Label: Is Part Of

Term description: The described resource is a physical or logical part of the referenced resource.

Guidelines for creation of content:

The isPartOf and hasPart relationships are essentially "parent/child" relationships--hierarchical in nature. With them can be expressed both one-to-one and one-to-many types of relationships.

hasPart

Label: Has Part

T*erm description:* The described resource includes the referenced resource either physically or logically.

Guidelines for creation of content:

See isPartOf for basic guidelines.

isReferencedBy

Label: Is Referenced By

Term description: The described resource is referenced, cited, or otherwise pointed to by the referenced resource.

Guidelines for creation of content:

The isReferencedBy and References refinements enable the expression of relationships that aid the user but are not necessary tied to the lifecycle or necessary for the intended use of the resource. This relationship might be used to link an article critical of a resource to that resource, a satire of a speech to the original speech, etc.

References

Label: References

Term description: The described resource references, cites, or otherwise points to the referenced resource.

Guidelines for creation of content:

See isReferencedBy for basic guidelines.

isFormatOf

Label: Is Format Of

Term description: The described resource is the same intellectual content of the referenced resource, but presented in another format.

Guidelines for creation of content:

This relationship is explicitly for the expression of relationships between resources for which format is the primary variable. Because Dublin Core maintains the principle of "one-to-one," each resource is expected to have its own description.

See also isVersionOf.

hasFormat

Label: Has Format

Term description: The described resource pre-existed the referenced resource, which is essentially the same intellectual content presented in another format.

Guidelines for creation of content:

See isFormatOf for basic guidelines.

conformsTo

Label: Conforms To

Term description: A reference to an established standard to which the resource conforms.

Guidelines for creation of content:

The standards referenced might be educational standards, accessibility standards, or any other established standard that is relevant to the use of the resource.

Refinement(s) for element: Coverage

Spatial

Label: Spatial

Term description: Spatial characteristics of the intellectual content of the resource.

Guidelines for creation of content:

Spatial characteristics may include geographic names, latitude/longitude, or other established georeferenced values. Clearly, this refinement does not allow complex or sophisticated georeferencing, but attention to standard schemes and controlled vocabularies should provide useful results. Controlled vocabulary terms can be drawn from recommended vocabularies, or standard labelling within the value can provide useful assistance to users and applications. For additional information on encoding spatial information see the DCMI Box Encoding Scheme and the DCMI Point Encoding Scheme.

Examples:

Spatial="Chicago, Ill."
Spatial="Lat: 44 00 00 S Long: 068 00 00 W Name: Patagonia"
Spatial="Upstate New York"

Temporal

Label: Temporal

Term description: Temporal characteristics of the intellectual content of the resource.

Guidelines for creation of content:

Temporal characteristics include those aspects of time that relate to the intellectual content of a resource and not its lifecycle. Examples might include a resource describing some aspect of the 19th century but itself created this year. In that case, the Temporal Coverage would be the 19th century, and the Date (or Date Created) would be 2003. Values can be text strings or encoded values. Specific suggestions for encoding Temporal characteristics may be found in the DCMI Period Encoding Scheme.

Examples:

Temporal="Jurassic Period"
Temporal="1922-1978"
Temporal="Twentieth Century"

Refinement(s) for element: Audience

Mediator

Label: Mediator

Term description: A class of entity that mediates access to the resource and for whom the resource is intended or useful. The audiences for a resource are of two basic classes: (1) an ultimate beneficiary of the resource, and (2) frequently, an entity that mediates access to the resource. The mediator element refinement represents the second of these two classes.

Guidelines for creation of content:

In an educational setting, a teacher might be designated the Mediator for a resource intended for use by a teacher in a classroom of students of a particular level or sharing other similar characteristics. Resources intended to be used directly by those same students would not include a Mediator. Mediators may be expressed in more or less specific terms, depending on the needs of the implementation. Controlled vocabularies can be useful in distinguishing Mediators.

Examples:

Mediator="Reading specialist"
Mediator="ESL teachers"

educationLevel

Label: Education Level

Term description: A general statement describing the education or training context. Alternatively, a more specific statement of the location of the audience in terms of its progression through an education or training context.

Guidelines for creation of content:

Commonly, this term would be used for a grade level for materials intended for an educational setting. Although no specific controlled vocabulary has been recommended for use with educationLevel, consistent use of terminology or reliance on an available controlled vocabulary enables more consistent results.

Examples:

educationLevel="elementary school students"
educationLevel="4th-5th grade"
educationLevel="secondary science"

Refinement(s) for element: Rights

accessRights

Label: Access Rights

Term description: Information about who can access the resource or an indication of its security status. Access Rights may include information regarding access or restrictions based on privacy, security or other regulations.

Guidelines for creation of content:

Access rights is intended to allow the characterization of restrictions to view, search or use resources, based on attributes of the resource itself or the class or category of user. An example would be a resource that was restricted to users holding a particular security clearance, or one that required login or authentication at a particular website.

Examples:

accessRights="Available to subscribers only."
accessRights="Viewable by Medium security cleared staff only."

license

Label: License

Term description: A legal document giving official permission to do something with the resource. Recommended best practice is to identify the license using a URI. Examples of such licenses can be found at http://creativecommons.org/licenses/.

Guidelines for creation of content:

License is designed to allow the inclusion of specific licensed uses to be specified. An example would be a resource that was available to be used freely but not for reproduction within commercial applications.

Examples:

license="http://creativecommons.org/licenses/by-nc-nd/2.0/legalcode"

license="Licensed for use under Creative Commons Attribution 2.0."

Refinement(s) for element: Identifier

bibliographicCitation

Label: Bibliographic Citation

Term description: A bibliographic reference for the resource. Recommended practice is to include sufficient bibliographic detail to identify the resource as unambiguously as possible, whether or not the citation is in a standard form.

Guidelines for creation of content:

Because this term is not describing a relationship to another resource, it should be limited to citations to the resource described in the remainder of the record. For instance, if the resource is an article for a journal, it is appropriate to include very specific information about the article, even page references, if such information is used to cite the article in a standard format for reference by other resources, *even if the article being described is in a digital format*.

Examples:

bibliographicCitation="ESOP, v.2, no. 1, Apr. 2003, p. 5-8"
bibliographicCitation="Nature, v.87, p. 200"

For additional guidance on using this refinement, see: Guidelines for Encoding Bibliographic Citation Information in Dublin Core (Proposed recommendation)

Valid XHTML 1.0! Valid CSS!

Metadata associated with this resource: http://dublincore.org/documents/usageguide/qualifiers.shtml.rdf

---
Date: Mon, 28 Feb 2005 13:59:21 +0000
From: Andy Powell <a.powell@UKOLN.AC.UK>
Subject: DCMES definitions - possible re-wording
To: DC-USAGE@JISCMAIL.AC.UK

On Friday, I was reminded of my action to look thru the current DCMES definitions and suggest possible revisions to make them more in line with the DCMI Abstract Model (DCAM).

Here are the ones that I find problematic:

Element Name: Format
Label: Format
Definition: The physical or digital manifestation of the resource.
Comment: Typically, Format may include the media-type or dimensions of the resource. Format may be used to identify the software, hardware, or other equipment needed to display or operate the resource. Examples of dimensions include size and duration. Recommended best practice is to select a value from a controlled vocabulary (for example, the list of Internet Media Types [MIME] defining computer media formats).

Problem: The current definition implies that the value is a "manifestation of the resource", which is not the intention. This issue has previously been raised on one of the the DCMI lists (I forget when and where) which suggests that at least one real end-user has mis-interpretted this wording in this way.

Proposed definition: The media-type or dimensions of the resource.
Proposed comment: Format may be used to identify the software, hardware, or other equipment needed to display or operate the resource. Examples of dimensions include size and duration. Recommended best practice is to select a value from a controlled vocabulary (for example, the list of Internet Media Types [MIME] defining computer media formats).

Element Name: Source
Label: Source
Definition: A Reference to a resource from which the present resource is derived.
Comment: The present resource may be derived from the Source resource in whole or in part. Recommended best practice is to identify the referenced resource by means of a string or number conforming to a formal identification system.

Problem: As per the DCAM, the value is the related resource, not a reference to the resource.

Proposed definition: A resource from which the present resource is derived.

Element Name: Relation
Label: Relation
Definition: A reference to a related resource.
Comment: Recommended best practice is to identify the referenced resource by means of a string or number conforming to a formal identification system.

Problem: same as with Source.

Proposed definition: A related resource.
Proposed comment: unchanged

Element Name: Rights
Label: Rights Management
Definition: Information about rights held in and over the resource.
Comment: Typically, Rights will contain a rights management statement for the resource, or reference a service providing such information. Rights information often encompasses Intellectual Property Rights (IPR), Copyright, and various Property Rights. If the Rights element is absent, no assumptions may be made about any rights held in or over the resource.

Problem: the comment refers both to a 'statement' and a reference to a service that provides a statement' This is an implementation issue anbd shouldn't be in the comment.

Proposed definition: unchanged
Proposed comment: Typically, Rights will contain a rights management statement for the resource. Rights information often encompasses Intellectual Property Rights (IPR), Copyright, and various Property Rights. If the Rights element is absent, no assumptions may be made about any rights held in or over the resource.

Comments by Pete

> Problem: As per the DCAM, the value is the related resource, not a
> reference to the resource.

[snip]

This same problem is also present in the comment for dc:description

Comment: Description may include but is not limited to: an abstract, table of contents, reference to a graphical representation of content or a free-text account of the content.

I think the "reference to" should be deleted (and for consistency an "a" added before "table of contents").

So:

Proposed Comment: Description may include but is not limited to: an abstract, a table of contents, a graphical representation of content or a free-text account of the content.

Re: Rights

> Proposed definition: unchanged
> Proposed comment: Typically, Rights will contain a rights management
> statement for the resource. Rights information often encompasses
> Intellectual Property Rights (IPR), Copyright, and various Property
> Rights. If the Rights element is absent, no assumptions may be made about
> any rights held in or over the resource.

This hadn't occurred to me before but after writing all that stuff about XML elements not being DC elements, I wondered whether the use of "contain" in the first sentence of the comment is also slightly misleading/inappropriate in the light of the DCAM.

XML elements are containers, and they have content.

But DC elements are not containers and do not have content: they are properties and they describe relationships between resources and values.

So it might be recast as something like

"Typically, a rights management statement for the resource is provided."

or

"Typically, rights information includes a rights management statement for the resource."

Date: Thu, 12 May 2005 10:41:26 +0100
From: Andy Powell <a.powell@UKOLN.AC.UK>
Subject: Vocabulary Encoding Scheme vs Syntax Encoding Scheme - summary
To: DC-USAGE@JISCMAIL.AC.UK

Summing up previous discussion, the test for whether a term is a syntax encoding scheme or a vocabulary encoding scheme is as follows:

- if the term defines a set of strings, either by enumerating them or by providing a rule for building them, the the term is a syntax encoding scheme

- if the term defines a class of values (not strings) , then the term is a vocabulary encoding scheme.

Therefore, we get the following:

Box	Syntax
DCMIType	Vocabulary
DDC	Vocabulary
IMT	Vocabulary
ISO3166	Vocabulary
ISO639-2	Vocabulary
LCC	Vocabulary
LCSH	Vocabulary
MESH	Vocabulary
Period	Syntax
Point	Syntax
RFC1766	Syntax
RFC3066	Syntax
TGN	Vocabulary
UDC	Vocabulary
URI	Syntax
W3CDTF	Syntax

Note that I've previously argued that RFC1766 and RFC3066 are vocabulary encoding schemes, but the RFCs specifically state that they define a set of 'tags', which I can't interpret in any other way than as an enumerated list of stings - therefore they are both syntax encoding schemes according to the above rule.

The key difference between syntax and vocabulary encoding schemes is their place in the abstract model. Syntax encoding schemes relate to 'value strings' - vocabulary encoding schemes relate the the 'value' (resource).