Title: DCMI Usage Board meeting agenda, Bath (UK)
Modified: 2004-02-25 15:44, Wednesday
Maintainer: Tom Baker
Latest version: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/
Description: Evolving agenda of topics for the next Usage Board meeting. The PDF meeting packet based on this page will be generated on 2004-02-25 and posted at http://www.bi.fhg.de/People/Thomas.Baker/Bath-meeting.pdf, at which time this agenda will be frozen.

NOTE: Additional material for the meeting
produced after 2004-02-25 will be listed at
http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/SUPPLEMENT/
and put in a supplementary packet to be posted at
http://www.bi.fhg.de/People/Thomas.Baker/Bath-meeting-supplement.pdf
shortly before the meeting.

2004-03-14, Sunday morning, 09:00-12:00 (150" plus 30" break)

TOPIC 1: DCMI Naming Policy (Tom, 15")
http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/docs-naming/
http://dublincore.org/usage/meetings/2004/03/naming-policy.html

TOPIC 2. Encoding schemes for Library Application Profile (Rebecca, 30")
http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/registration-proposals/
http://dublincore.org/resources/faq/

TOPIC 3: Registration of Vocabulary Encoding Schemes (Traugott, 60")
http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/registration/
http://www2.elsevier.co.uk/~tony/info/info.html

http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/CACHE/Weibel.InfoURI.Registry.pdf

2004-03-14, Sunday afternoon, 13:00-18:00 (240" plus 60" break)

TOPIC 4. Proposed terms for a Collection Description profile (Andrew, 60")
http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/terms-collection/
http://www.ukoln.ac.uk/metadata/dcmi/collection-provenance/
http://www.ukoln.ac.uk/metadata/dcmi/collection-isAvailableAt/

TOPIC 5. Scope of DCMI Namespaces (Andy, Pete Johnston as guest, 30")
http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/dcmi-namespaces/
http://www.ukoln.ac.uk/metadata/dcmi/dcmi-nmspc-scope/20040222/

TOPIC 6. DCMI Abstract Model (Andy, 60")
http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/model/
http://www.ukoln.ac.uk/metadata/dcmi/abstract-model/
http://dublincore.org/usage/documents/principles/

TOPIC 7. Guidelines for DC Application Profiles (Tom, 30")
http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/docs-DCAPs/

http://dublincore.org/usage/meetings/2004/03/cwa14855-20040210.pdf
http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/CACHE/DCMI-CEN-agreement.pdf

TOPIC 8. PBCore Metadata Dictionary (Diane, 30")
http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/profiles-pbcore/

TOPIC 9. Review of Application Profiles (Tom, 30")
http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/profiles/
http://dublincore.org/usage/documents/profiles/
http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/CACHE/Process6.html

2004-03-15, Monday morning, 09:00-12:00 (150" plus 30" break)

TOPIC 10. AskDCMI (Diane, 30")
http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/askdcmi/
http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/CACHE/AskDCMI.txt

TOPIC 11. Proposals for dc:rights-related terms (Andrew, 60")
http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/terms-rights/
http://www.ukoln.ac.uk/metadata/dcmi/dc-rights/

2004-03-15, Monday afternoon, 13:00-16:00 (150" plus 30" break)

TOPIC 12. MARC Relator terms and dc:contributor (Rebecca, 45")
http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/terms-relators/
http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/CACHE/Agent-Roles-Guidelines2.txt

TOPIC 13. Dublin Core and OAI (Tom, 30")
http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/oai/

TOPIC 14. Unfinished business (Tom, Diane, Stuart, 15")
http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/etc-old/

TOPIC 15. Usage Board issues (Tom, 30")
http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/usageboard/
http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/CACHE/DC-2004-call.txt

Title: DCMI Usage Board meeting agenda, Bath (UK) - Supplementary material
Modified: 2004-02-25 15:44, Wednesday
Maintainer: Tom Baker
Latest version: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/SUPPLEMENT/
See also: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/
Description: Evolving agenda of topics for the next Usage Board meeting. The main PDF packet for the meeting was posted on 2004-02-25. Materials produced after 2004-02-25, listed here, will be put in a supplementary packet to be posted on 2004-03-11 at http://www.bi.fhg.de/People/Thomas.Baker/Bath-meeting-supplement.pdf.

TOPIC 3: Registration of Vocabulary Encoding Schemes
http://wip.dublincore.org/schemes/index.html
http://www.lub.lu.se/~traugott/drafts/vocab-scheme-Jan04.html
http://www.lub.lu.se/~traugott/drafts/vocab-guide6.html

TOPIC 8. PBCore Metadata Dictionary
http://www.utah.edu/cpbmetadata/PBCore/
http://www.utah.edu/cpbmetadata/PBCore/Title.html
http://www.utah.edu/cpbmetadata/PBCore/TitleProgram.html
http://www.utah.edu/cpbmetadata/PBCore/TypeForm.html
http://www.utah.edu/cpbmetadata/PBCore/RelationType.html
http://www.utah.edu/cpbmetadata/PBCore/RelationIdentifier.html
http://www.utah.edu/cpbmetadata/PBCore/CoverageTemporal.html
http://www.utah.edu/cpbmetadata/PBCore/AudienceRating.html
http://www.utah.edu/cpbmetadata/PBCore/Creator.html
http://www.utah.edu/cpbmetadata/PBCore/CreatorRole.html
http://www.utah.edu/cpbmetadata/PBCore/RightsUsage.html
http://www.utah.edu/cpbmetadata/PBCore/RightsReproduction.html
http://www.utah.edu/cpbmetadata/PBCore/RightsAccess.html
http://www.utah.edu/cpbmetadata/PBCore/DateAvailableStart.html
http://www.utah.edu/cpbmetadata/PBCore/FormatIdentifier.html
http://www.utah.edu/cpbmetadata/PBCore/FormatI_ChannelConfig.html
http://www.utah.edu/cpbmetadata/PBCore/FormatTimeStart.html
http://www.utah.edu/cpbmetadata/PBCore/FormatType.html
http://www.utah.edu/cpbmetadata/PBCore/FormatEncoding.html
http://www.utah.edu/cpbmetadata/PBCore/Identifier.html
http://www.utah.edu/cpbmetadata/PBCore/Language.Usage.html
http://www.utah.edu/cpbmetadata/PBCore/Location.html

TOPIC 11. Proposals for dc:rights-related terms
http://www.openarchives.org/documents/OAIRightsWhitePaper.html

TOPIC 12. MARC Relator terms and dc:contributor
http://www.loc.gov/marc/dc/marcrelcodes.rdf

APPENDIX Reference materials
http://dublincore.org/usage/meetings/2004/03/2003-09-27.Minutes-seattle-

final.txt
http://dublincore.org/usage/documents/process/

TOPIC 1: DCMI Naming Policy
TOPIC 2. AskDCMI
TOPIC 4. Proposed terms for a Collection Description profile
TOPIC 5. Scope of DCMI Namespaces
TOPIC 6. DCMI Abstract Model
TOPIC 7. Guidelines for DC Application Profiles
TOPIC 9. Review of Application Profiles
TOPIC 10. Encoding schemes for Library Application Profile
TOPIC 13. Dublin Core and OAI
TOPIC 14. Unfinished business
TOPIC 15. Usage Board issues

Title: "DCMI Naming Policy" issue
Modified: 2004-02-25 15:44, Wednesday
Maintainer: Tom Baker
Latest version: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/docs-naming/
See also: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/

SUMMARY

I have extensively rewritten an earlier draft by Stu Weibel (see below), renaming it "DCMI Naming Policy" and posting it at:

http://dublincore.org/usage/meetings/2004/03/naming-policy.html

In Bath, I would like to ask the Usage Board to indicate its approval of this document (or conditional approval, with comments, as the case may be). We would then report this approval to the Directorate, which would formally approve the document as a DCMI Recommended Resource.

BACKGROUND

As originally declared in 1998, the fifteen elements of Dublin Core had "names" (which called "labels" at the time) in uppercase [1]. Although there was discussion about changing the names to lowercase (notably at DC-8 in Ottawa, October 2000), this was not done, and the uppercase names progressed through the NISO and then ISO processes and are currently shown in ISO 15836 of February 2003 [2]. The uppercase names are thus also reflected in the DCMI document "Dublin Core Metadata Element Set, Version 1.1: Reference Description", which is word-for-word identical with ISO 15836 [3].

However, the official DCMI term declarations -- those that have a Usage Board status of "Recommended" (as reflected in [4], [5], and [6]) -- currently have URIs based on a name in lowercase. This change was based on a decision of the Directorate dated 2002-09-13 (as reported on the Web page "DCMI Usage Board Decisions" [7]) which, however, points to a placeholder document [8].

In September, Andy posted the most recent draft of that paper on the UKOLN Web site [9] and agreed to shepherd the case change proposal through UB for eventual publication as a DCMI recommendation (see point A.1 below). Responsibility for the text, however, lies with the currently listed author, Stu Weibel, or at any rate with the DCMI Directorate.

After the case change is approved, the Directorate would presumably seek a corresponding change in the corresponding formal standards documents. In April 2002, John Kunze posted a corrected draft of RDF 2413 [10] but this has since expired and been replaced by a placeholder [11].

[1] http://www.ietf.org/rfc/rfc2413.txt
[2] http://www.niso.org/international/SC4/n515.pdf
[3] http://dublincore.org/documents/dces/
[4] http://dublincore.org/documents/dcmi-terms/
[5] http://dublincore.org/usage/terms/history/
[6] http://dublincore.org/2003/03/24/dces#
[7] http://dublincore.org/usage/decisions/#Decision-2002-02
[8] http://dublincore.org/documents/2002/09/13/case/
[9] http://www.ukoln.ac.uk/metadata/dcmi/case-policy/, archived as:
http://dublincore.org/usage/meetings/2004/03/2003-09-28.case-policy.html
[10] http://www.ietf.org/internet-drafts/draft-kunze-rfc2413bis-01.txt
[11] http://www.ietf.org/internet-drafts/draft-kunze-rfc2413bis-02.txt

---

APPENDIX: Recent decisions and action items

A.1. SEATTLE Action Item 19: Andy to shepherd the case change proposal through UB for eventual issue as a DCMI recommendation.

Title:	**DCMI Policy on Naming Terms**
Creator:	Stuart Weibel
Creator:	Thomas Baker
Date Modified:	2004-02-05
Identifier:	http://dublincore.org/documents/2004/03/15/naming/
Is Replaced By:	Not applicable
Latest Version:	http://dublincore.org/documents/naming/
Status of Document:	This is a DCMI Recommended Resource.
Description of Document:	This document describes the policy followed in assigning Names to DCMI terms, particularly with regard to case sensitivity.

DCMI Term Names

The Dublin Core Metadata Initiative maintains sets of metadata terms. Each metadata term is assigned a *name* -- a character string or "token" that is unique in the context of a particular DCMI term set [DCMI-TERMS]. This string is invariant across encodings and languages. In accordance with the DCMI Namespace Policy, the *name* of a term is appended to the URI of a DCMI namespace to construct a globally unique identifier (URI) for that particular term [DCMI-NAMESPACE]. The typology of DCMI metadata terms is described in the DCMI Grammatical Principles [DCMI-PRINCIPLES].

Case Policy for DCMI Term Names

Names of DCMI Elements and Element Refinements start with lowercase characters but may contain uppercase characters where a term name is comprised of multiple concatenated words. In such cases, the leading character of additional words will be capitalized to improve human-readable clarity. Examples include:

creator
audience
isReplacedBy

which are identified with the following URIs:

http://purl.org/dc/elements/1.1/creator
http://purl.org/dc/terms/audience
http://purl.org/dc/terms/isReplacedBy

Names of Encoding Schemes identifying controlled vocabularies by acronym are represented in all uppercase characters. Examples include:

DDC
W3CDTF
ISO639-2

which are identified with the following URIs:

http://purl.org/dc/terms/DDC
http://purl.org/dc/terms/W3CDTF
http://purl.org/dc/terms/ISO639-2

Names of Encoding Schemes other than acronyms are represented with a leading uppercase character followed by lowercase characters. Examples include:

Period
Box

which are identified with the following URIs:

http://purl.org/dc/terms/Period
http://purl.org/dc/terms/Box

Names of values within a DCMI-maintained controlled vocabulary such as the DCMI Type Vocabulary begin with an uppercase character followed by lowercase, but with subsequent concatenated words in the name capitalized as well. Examples include:

InteractiveResource
Collection
Dataset

which are identified with the following URIs:

> http://purl.org/dc/dcmitype/InteractiveResource
> http://purl.org/dc/dcmitype/Collection
> http://purl.org/dc/dcmitype/Dataset

No DCMI Term Names will be assigned that differ from other Names only in regard to case.

Historical note

Since the mid-1990s, Dublin Core has developed in parallel with World-Wide Web technology, and Dublin Core metadata has been deployed using all of the major encoding syntaxes that have evolved. In the transition from HTML to XML, XHTML, and RDF/XML, conventions regarding the naming of metadata elements have consolidated in the wider Web community, and DCMI policy has changed to follow the emerging model of good practice.

In 1998, the Dublin Core element set was published using Names (at the time called Labels) which had a leading uppercase character, e.g. Title and Creator [RFC2413]. In October 2000, the DCMI Advisory Committee decided to change the Names of elements to lowercase in order to bring DCMI practice into line with conventions widely (though not universally) followed in existing Dublin Core applications and in the XML and RDF/XML communities more generally.

Unfortunately, this decision was propagated throughout DCMI documentation with some delay [DC-ELEMENTS]. In the meantime, the Dublin Core Metadata Element Set had progressed through formal standardization channels for recognition first as NISO Z39.85-2001 and CEN Workshop Agreement CWA 13874, then as ISO 15836 -- but with element Names starting in uppercase [ISO15836].

The DCMI Directorate is not aware of any applications or implementations where this inconsistency with regard to the case of element Names has caused any practical problems. For the sake of consistency, however, the Directorate will undertake to amend existing specifications as permitted by their maintenance cycles.

Confusion over case sensitivity is commonplace in Web protocols, and metadata application developers are likely to find many permutations of case in instance data and term declarations. Thus, to paraphrase a dictum from early Web protocol development, applications should:

> *...be rigorous in what you export, and tolerant in what you import from other applications.*

In practice, this suggests that applications are well advised to normalize case when parsing terms for identity comparisons. Prudence mitigates against the use of case to distinguish between alternative

identities of related terms in any namespace, and it is DCMI policy that such distinctions not be made within its own namespaces, so it is unlikely that errors would be introduced by normalizing case.

References

[DCMI-TERMS]	http://dublincore.org/documents/dcmi-terms/
[DCMI-NAMESPACE]	http://dublincore.org/documents/dcmi-namespace/
[RFC2413]	http://www.ietf.org/rfc/rfc2413.txt
[DC-ELEMENTS]	http://dublincore.org/documents/2002/10/06/current-elements/
[ISO15836-2003]	http://www.niso.org/international/SC4/n515.pdf

Feedback on this document

The DCMI Directorate welcomes comments and suggestions on this policy: dcmi-feedback@dublincore.org

---

Metadata associated with this resource: http://dublincore.org/documents/case-policy/index.shtml.rdf

Topic: Encoding schemes for the Library Application Profile
Modified: 2004-02-25 15:44, Wednesday
Maintainer: Tom Baker
Latest version: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/registration-proposals/
See also: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/

Shepherd: Rebecca Guenther

SUMMARY (Tom)

In Ithaca, and then again in Seattle, we discussed whether the following Encoding Schemes, needed for the DCMI Library Application Profile, should be formally approved and declared by DCMI:

1) http://purl.org/dc/terms/ISO8601
2) http://purl.org/dc/terms/AAT
3) http://purl.org/dc/terms/TGM2
4) http://purl.org/dc/terms/SICI
5) http://purl.org/dc/terms/ISBN
6) http://purl.org/dc/terms/ISSN
7) http://purl.org/dc/terms/DOI
8) http://purl.org/dc/terms/GAC
9) http://purl.org/dc/terms/MARC Country Codes (sic)

Whether or not this should be done is a question that differs case by case, depending on whether the maintainers of the resources in questions were likely themselves to assign URIs or whether evolving practice for URIs and URNs would likely provide workable solutions in the short term that could make a DCMI-specific approach redundant.

The following is my rough-cut assemblage of notes and excerpts from email as a guide; Rebecca will provide the latest status on each.

Incidentally, I discovered while writing this summary that clicking on any of the URIs above takes one to the top of the file http://dublincore.org/2003/03/24/dcq, even though none of those URIs exist...

IN DETAIL

In Ithaca in June 2004, we discussed proposals for several encoding schemes [1].

The Ithaca notes show we approved two proposals [2]:
-- http://purl.org/dc/terms/ISO8601 (by a vote of 7 to 1)
-- http://purl.org/dc/terms/AAT (by a vote of 8 to 0)

We also decided [2]: "New terms to go in the terms namespace with one or two exceptions (MARC, DOI?). Discussions and agreement on new terms to be continued online in early to mid-July."

I see the proposals were discussed in July 2004 [3,4,5] and then in Seattle, and this would appear to be the state of affairs:

1) http://purl.org/dc/terms/ISO8601
   Discussed and approved. I have put a draft of the decision text at http://dublincore.org/usage/meetings/2004/03/2004-01.ISO8601.txt. If Rebecca confirms this is accurate, we can publish it after the Bath meeting.

   Note that Roland suggested [4]: "As far as ISO 8601 is concerned i refer to the Abstract of http://www.w3.org/TR/NOTE-datetime as well as http://www.w3.org/TR/xmlschema-2/#isoformats May I suggest to include both links with the registration of ISO 8601." Rebecca agreed that these links should be included.

2) http://purl.org/dc/terms/AAT
   In Ithaca, we decided that Rebecca needed to investigate whether AAT has a persistent URI at Getty. So it was deferred for further research (the URI could be added later once approved). This should now change to use the MARC namespace. Getty has not established persistent URIs. They were comfortable with using the MARC source code rather defining a new one in the DC namespace. (Contact there was Murtha Baca). I included this revised proposal in the document discussed in Seattle and it was approved. The URI should be: http://www.loc.gov/marc.source/aat. There was some discussion of whether, as an Encoding Scheme, AAT would need to be in uppercase, but it was affirmed that such URI assignment policies are the prorogative of the agency assigning the URIs.

3) http://purl.org/dc/terms/TGM2
   In Ithaca, we decided this needs to be revised and resubmitted for a vote [3]. In Seattle, this was submitted as "gmgpc" and using the MARC namespace (the same as TGM2). So it is: http://www.loc.gov/marc.source/gmgpc Same question about upper vs. lower case.

4) http://purl.org/dc/terms/SICI
5) http://purl.org/dc/terms/ISBN
6) http://purl.org/dc/terms/ISSN
7) http://purl.org/dc/terms/DOI
   Rebecca in July [3]: "Andy indicated that any of these may be expressed as a URI, and that they should be expressed this way. Thus, it is not necessary to add, since the

encoding scheme "uri" would be indicated. (I think there is disagreement about this approach, since some have not been registered through IANA as a URI scheme or URN name space.) Comments?"

In February 2004: "Can't be expressed as URI, since it hasn't been approved as a URN namespace or URI scheme. See below."

Roland on SICI [4]: "Some Problem with SICI: The Contribution Segment seems to explicitly use < and > signs. Use of SICI's in a literal way could cause trouble in XML. Need to get into touch with SICI about standardized escaping mechanism - seeAlso http://sunsite.berkeley.edu/SICI/version2.html (on a related problem with interpreting SICI as URN)"

Rebecca, 2004-02: This could still be a problem and needs to be done.

SEATTLE Vote on SICI [6]: UB approves SICI conditionally (and unanimously). If the IETF InfoURI proposal is accepted SICI will be encoded as a URI and UB approval for SICI as an encoding scheme is unnecessary. If this is the case the vote becomes redundant.

Roland on ISBN [4]: "ISBN registered as Formal URN Namespace: seeAlso RFC3187"
Rebecca, 2004-02: Yes. Express as URI.

Roland on ISSN [4]: "ISSN registered as Formal URN Namespace: seeAlso RFC3044"
Rebecca, 2004-02: Yes. Express as URI.

Roland on DOI [4]: doi:10.1004/123456. -- But Tom does not believe this prefix was approved by IETF.

Roland: Quoted from http://www.doi.org/faq.html : "A DOI is an implementation of URI.." seeAlso NISO Z39.84 [I don't know whether Z39.84 has been recognized outside the US] - and info:doi/

Rebecca: But it's not registered as a URI scheme. That is essentially why info came about. When those things were written I think noone knew what a stumbling block this was going to be.

SEATTLE Vote on DOI [6]: UB approves DOI conditionally (and unanimously). If the IETF InfoURI proposal is accepted DOI will be encoded as a URI and UB approval for DOI as an encoding scheme is unnecessary. If this is the case the vote becomes redundant.

Rebecca, 2004-02: On SICI and DOI: Info has not been formally approved by IETF, but NISO has already established a registry

for info schemes. Does that give it some sort of status? I would think so. We could then bring a proposal to approve info in the NISO namespace and use schemes that are registered there.

Roland: There was quite some discussion about info on rdf-interest and confusion. We need to consider what to do with pending (such as info) and possibly conflicting standardization approaches (such as doi).

Rebecca: DOI is registered under INFO, so is not a conflicting approach. See: http://info-uri.info/

Roland: Right. But NISO has set up the registry anyhow. My point was that you had said SICI would come under info but that DOI was going another way. But that is not the case--they are both going the route of info.

Andy, 2004-02-23:

1) A URI scheme doesn't have to be registered for something to be a valid URI - it just has to conform to the URI spec. So
   doi:1.100/23435
   is a valid URI irrespective of whether the 'doi' scheme is registered with IANA or not.

2) It is not clear to me (based on discussions with Norman Paskin at DOI) whether the IDF is going firmly down the 'info' route or not. It is still the case that they only really recommend the doi:10.100/234 form of encoding AFAIK. So, 'info' may be one way of encoding a DOI, but it may be that the IDF never really endorse this mechanism??

   However, this is largely irrelevant... in order to encode a DOI in DC metadata you only need to use the URI encoding scheme, either as

   <meta name="DC.identifier" scheme="DCTERMS.URI" content="doi:10.1000/234"/> or <meta name="DC.identifier" scheme="DCTERMS.URI" content="info:doi/10.1000/234" />

   I have to confess that I still don't understand why we approved DOI as an encoding scheme in Seattle. Someone kindly put me out of my misery and remind we why we want to invent a DC-specific way of encoding DOIs (and the rest!)?

8) http://purl.org/dc/terms/GAC
9) http://purl.org/dc/terms/MARC Country Codes (sic)
   To be revised and resubmitted for a vote.

SEATTLE Decision [6]: Rebecca will consult with Andy as to

proper URI identification of the Marc GAC and MARC Country Codes schemes. They will both live in a LC namespace.

These go in the LC namespace:
-- http://www.loc.gov/marc.gac/
-- http://www.loc.gov/marc.countries

REFERENCES

[1] http://dublincore.org/usage/meetings/2003/06/dclib-encodingschemes.html
[2] http://www.jiscmail.ac.uk/cgi-bin/wa.exe?A2=ind0309&L=dc-usage&T=0&O=D&P=773
[3] http://www.jiscmail.ac.uk/cgi-bin/wa.exe?A2=ind0307&L=dc-usage&T=0&F=&S=&P=4409
[4] http://www.jiscmail.ac.uk/cgi-bin/wa.exe?A2=ind0307&L=dc-usage&T=0&F=&S=&P=4409
[5] http://www.jiscmail.ac.uk/cgi-bin/wa.exe?A2=ind0307&L=dc-usage&T=0&F=&S=&P=4669
[6] http://dublincore.org/usage/meetings/2004/02/2003-09-27.Minutes-seattle-final.txt

DCMI Frequently Asked Questions (FAQ)

What is metadata?

The simplest definition of metadata is " structured data about data."

Metadata is descriptive information about an object or resource whether it be physical or electronic. While metadata itself is relatively new, the underlying concepts behind metadata have been in use for as long as collections of information have been organized. Library card catalogs represent a well-established type of metadata that has served as collection management and resource discovery tools for decades.

Metadata can be generated either "by hand" or derived automatically using software.

Where is the metadata on the Dublin Core website?

We use separate .rdf files that contain the Dublin Core metadata information rather than including the metadata in the HTML. If you look at most of the pages on the DCMI Website, you'll see a link to the metadata at the bottom of each web page.

What is a resource?

In Web terminology, a resource is "anything addressable via a URL." However, Dublin Core implementations are not necessarily Web-based.

Dublin Core metadata can be used to describe any kind of resource - including various collections of documents and non-electronic forms of media such as a museum or library archive.

What is resource discovery?

A card catalog is the most common physical representation of a metadata system and is a perfect example of a system originally devised for locating resources that evolved into its own information system.

What is the Dublin Core Metadata Initiative?

The Dublin Core Metadata Initiative (DCMI) is an organization dedicated to fostering the widespread adoption of interoperable metadata standards and promoting the development of specialized metadata vocabularies for describing resources to enable more intelligent resource discovery systems.

The first Dublin Core Series Workshop took place in Dublin, Ohio in 1995. Since that time, the DCMI has been committed to the continual refinement of a "core" foundation of property types and values to provide vertically specific (semantic)information about Web resources, much in the same way a library card catalog provide indexes of book properties.

What is the Dublin Core?

Dublin Core metadata is used to supplement existing methods for searching and indexing Web-based metadata, regardless of whether the corresponding resource is an electronic document or a "real" physical object.

The Dublin Core Metadata Element Set (DCMES) was the first metadata standard deliverable out of the DCMI was an IETF RFC 2413. DCMES provides a semantic vocabulary for describing the "core" information properties, such as "Description" and "Creator" and "Date".

Dublin Core metadata provides card catalog-like definitions for defining the properties of objects for Web-based resource discovery systems.

What is the Dublin Core Metadata Element Set?

The Dublin Core Metadata Element Set is a set of 15 descriptive semantic definitions. It represents a core set of elements likely to be useful across a broad range of vertical industries and disciplines of study.

The Dublin Core Metadata Element Set was created to provide a core set of elements that could be shared across disciplines or within any type of organization needing to organize and classify information.

Who can benefit from using Dublin Core metadata?

Anyone can use Dublin core metadata to describe the resources of an information system. Web pages are one of the most common types of resources to utilize the Dublin Core's descriptions, usually within HTML's meta tags however increasingly there are many digital archives of physical objects that are starting to make use of the Dublin Core.

Dublin Core metadata is being used as the basis for descriptive systems by several community interest groups such as:

- educational organizations
- libraries
- government institutions
- scientific research sector
- Web page authors
- businesses requiring more searchable sites
- corporations with vast knowledge management systems

(See also our listing of Dublin Core-based projects).

What is the difference between "Simple" ("unqualified") and "Qualified" Dublin Core?

"Simple Dublin Core" is Dublin Core metadata that uses no qualifiers; only the main 15 elements of the Dublin Core Metadata Element Set are expressed as simple attribute-value pairs without any "qualifiers" (such as encoding schemes, enumerated lists of values, or other processing clues) to provide more detailed information about a resource.

"Qualified Dublin Core" employs additional qualifiers to further refine the meaning of a resource. One use for such qualifiers are to indicate if a metadata value is a compound or structured value, rather than just a string.

Qualifiers allow applications to increase the specificity or precision of the metadata. They may also introduce complexity that could impair the metadata's compatibility with other Dublin Core software applications. With this in mind, designers should only select from the set of approved Dublin Core qualifiers that were developed by the Dublin Core community process.

Unfortunately, qualifiers often introduce additional complexity that can make metadata less interoperable unless approved DC Qualifiers developed within the DCMI are used with such interoperability considerations in mind.

A "date" is one example of a DC element that has the option of being further specified to identify it as a particular kind of date (date last modified, date published, etc.).

The use of a controlled vocabulary, such as Dewey Decimal Classification, is another method that could be used to further "qualify" the meaning of resource.

For examples of embedding qualified DC in HTML please read the "Qualified HTML Examples" section of the "Using Dublin Core" usage guide.

The Australian Government Locator Service is a good example of a major implementation using Dublin Core metadata

that has added four local elements to Dublin Core in the proper, recommended way.

How do I begin implementing the Dublin Core Metadata Element set and where can I ask questions if I have questions regarding my implementation?

This is one of the most often asked questions and perhaps the most difficult to provide a catch-all answer. The wise ones say, "It depends." Unfortunately, that is the truth. It depends on your hardware, software, system resources and the goals you are trying to achieve. In a nutshell, we have a Usage Guide which provides some insight. Beyond that, we highly recommend posing questions to the DC-General mailing list. Most questions can be answered there or redirected to a place where the answer can be found.

What is the relationship between the DCMI and other Internet Standards groups?

The Dublin Core Metadata Initiative (DCMI) is a consensus building organization that has relationships with many standards activities. A number of people in the DCMI are active in the World Wide Web Consortium (W3C) (DC is the prototype application that drove the development of the Resource Description Framework, or RDF in the W3C).

Our own standardization activities take place in the IETF (RFC 2413 is reference description of the initial version of the Dublin Core), and there are currently formal DC standardization activities underway in CEN (the European Information Industry Standardization Forum) and in NISO (the North American Information Standardization Organization) and the IEEE (Institute of Electrical and Electronics Engineers).

Was Dublin Core metadata designed to be used only to describe digital and Web-based resources?

No. The scope of the Dublin Core was specifically designed to provide a metadata vocabulary of "core" properties able to provide basic descriptive information about any kind of resource, regardless of the media format, area of specialization or cultural origin. It is important that a semantic model used for resource discovery is not dependent on the medium of the resource it means to describe.

The Dublin Core metadata vocabulary is the result of many years of collaborative research to determine a common set of properties universal for describing any type of resource. The use of a standardized general classifications system also enables metadata of such collections to be combined and for knowledge contained within each collection to be shared.

Since most Dublin Core implementations only need to process a resource's descriptive metadata, the medium of that resource becomes a non-issue. This enables DC metadata to be used by museums and other organizations interested in cataloging specialized types of mixed-media collections, while maintaining an open framework preserving their ability to share metadata with other DC implementors.

How is Dublin Core metadata used?

The term "Dublin Core metadata" is usually in reference to the 15 elements of the Dublin Core Metadata Element Set (DCMES) and its limited set of optional approved qualifiers tested and approved as "safe" for use by the Dublin Core community.

The DCMES was published in 1998 as an IETF RFC 2413. Dublin Core metadata is used to supplement existing methods for searching and indexing Web-based resources, providing a semantic vocabulary to describing the "core" properties of a resource object (such as "Description" and "Creator" and "Date").

How is Dublin Core metadata stored?

Dublin Core metadata is often stored as name-value pairs within META tags, which are placed within the HEAD elements of an HTML document.

However, it can also be located in an external document or loaded into a database enabling it to be indexed and manipulated from within a propriety application.

How can I embed Dublin Core metadata within my HTML documents?

Dublin Core metadata can be stored using the meta element in the head of HTML documents. An informational IETF RFC (2731) titled "Encoding Dublin Core Metadata in HTML" defines the standard way to do this. A DCMI Note describes one method for storing Qualified DC in HTML.

What is the relationship between Dublin Core Metadata and RDF and XML?

Dublin Core Metadata Element Set (IETF RFC 2413) and RDF (Resource Description Framework) (W3C Recommendation) are two distinct specifications. Neither requires the other, but their co-evolution forms a natural complement within the Web's greater metadata architecture.

Both the Dublin Core and RDF communities have a number of members in common, and have evolved side-by-side. The Dublin Core community provided much of the basic requirements that were used to design RDF. In turn, the development of RDF provided the Dublin Core community with a much more formal underlying data model that has helped it to determine best practices and universal solutions (rather than ad hoc Band-Aids) for many of the detailed problems that were encountered during the deployment process.

Below is an example of how the Dublin Core vocabulary can be used to define additional semantics about the resources described within an RDF fragment:

```
<?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 about="http://purl.org/DC/documents/notes-cox-816.htm">
    <dc:title>Recording qualified Dublin Core metadata in HTML</dc:title>
    <dc:description> We describe a notation for recording
    qualified Dublin Core metadata in HTML meta elements.  The syntax
    includes recommended usage of the standard HTML syntax to record
    the different classes of qualification needed to represent the
    model.</dc:description>
    <dc:date>1999-08-18</dc:date>
    <dc:format>text/html</dc:format>
    <dc:language>en</dc:language>
    <dc:publisher>Dublin Core Metadata Initiative</dc:publisher>
  </rdf:Description>
</rdf:RDF>
```

What is the Dublin Core data model?

The basic Dublin Core data model is defined by its 15 elements and the relationships defined between the resource-of-interest and whatever other resource is "in-scope" for DC.

In particular, the "Relation" and "source" elements are used to indicate a connection with another resource of any type. "Creator", "Contributor" and "Publisher" elements relate the present resource to a party who has some responsibility for it. The "Coverage" element relates the present resource to a place or to a time-period. The value recorded for each of these elements, therefore, should normally be strictly an identifier of another resource, which could have its own

DCMES description.

A completely abstracted DC data model must also include its two types of qualifiers. Value qualifiers (which store an identifier for the vocabulary, encoding or language of the value) and element qualifiers, which are used to further refine the semantic meaning of an element.

However, many users have found it useful to add extra information to Dublin Core descriptions. Particularly when values are frequently drawn from a controlled vocabulary (e.g. a keyword list, such as Library of Congress Subject Headings, LCSH), or written using a special notation (e.g. the ISO8601 format for dates and times).

Other techniques include using a particular natural language (in the case of values written in text-strings), which would make the information more useful if a client were informed of the source for this vocabulary, the definition of the notation, or the name of the language used.

How do I participate in a discussion about the Dublin Core?

Anyone may participate in discussions about Dublin Core metadata by simply joining the appropriate mailing list for the working group activity of interest.

The DC-General mailing list is the general forum for community participation and submitting general feedback.

What is the maximum length for each field in Dublin Core?

There are no limits to field length.

What search-engines support the Dublin Core Metadata Element Set?

Several commercial and non-commercial search engines will index META elements with just a little configuration.

A recent inquiry on the dc-general mailing list produced this list:

- Ultraseek
- Swish-E
- Microsoft's Index Server
- Blue Angel Technologies MetaStar
- Verity Search 97 Information Server

To get a good overview of what software is out there see Search Tools and Search Engine Watch.

The well-known "all the Web" search engines including AltaVista, Yahoo, HotBot, etc. tend to avoid using the information found in meta elements in their indexing. This is because, unless the pages are from guaranteed "trusted" servers, the meta information is commonly used by unscrupulous content-providers for spamming, to mislead the indexes into givingWeb-pages a misleading rating.

What is an attribute-value pair?

Attribute-value pairs are used within Dublin Core metadata to represent the properties of a resource or object.

Information such as "Author", "Creator" and "Date" are all examples of Dublin Core elements that are implemented as attribute-value pairs within HTML's META element to provide additional semantics about a resource.

Web resources are effectively information objects complete with properties that can be expressed in any number of ways. The Dublin Core Metadata Element Set (DCMES) enables properties to be assigned to a Web resource the exact

same way we might fill in the blanks for a card catalog record.

What is the Warwick Framework?

The Warwick Framework is a set of design principles that have guided the development of the Dublin Core since the Second Dublin Core Metadata Workshop in Warwick, UK.

The Warwick provides a metadata-based school of thought that believes different kinds of metadata can be used to describe the same resource in disparate ways to accomplish different goals.

For more information about the Warwick Framework, please read "The Warwick Metadata Workshop: A Framework for the Deployment of Resource Description".

Can I add a new element to Dublin Core?

In theory, yes. A DCMES element specified with an element qualifier is, effectively, a "new" element or property- with a more specialized meaning than its parent element.

However, it is not possible to create a new Dublin Core element whose meaning goes beyond the scope of of the original elements in the DCMES.

It is expected that local or application-specific requirements may require additional qualifiers or elements that have not been approved by the Dublin Core community at large. Nevertheless, designers should employ additional qualifiers with both caution and the understanding that interoperability could suffer as a result.

In cases where additional qualifiers are being utilized, it may be helpful to bring this to alert the Dublin Core Directorate in order to promote the wider use of such qualifiers.

For examples of embedding qualified DC in HTML please read the "Qualified HTML Examples" section of the "Using Dublin Core" usage guide.

What is the Open Metadata Registry?

The DCMI's Open Metadata Registry is a Web-based semantic modeling tool that uses a form-driven user interface to enable an end-user to define relationships between different vocabularies.

The semantic framework for the application was created using Dublin Core metadata and RDF Schemas.

How do I store proper names in Dublin Core metadata?

The encoding of personal names is a difficult task within most metadata systems, and DC is no exception. The difficult part is that naming conventions tend to vary from culture to culture.

The recommendation for Dublin Core metadata is to encode the family name first, which supports effective collation of names and is consistent with most naming conventions globally.

What is the "Dumb-Down" Principle?

The so called "Dumb-Down Principle" simply means that in any use of a qualified DC element, the qualifier may be dropped and the remaining value of the element should still be a term that is useful for discovery.

For example, there are several date qualifiers that might be used to enhance the precision of various dates associated

with a resource. Dropping the date qualifier (say, for example, Date-Created) will still leave a useful date for discovery, though perhaps not quite as useful as if the qualifier were included.

Similarly, the specification of a subject term from LCSH, for example, is still useful even if one does not know it was selected from a controlled vocabulary.

The basic idea is that qualifiers should improve the precision of a piece of metadata, but the metadata should still be useful even without that extra precision (that is, dropping the qualifier has 'dumbed-down' the metadata).

How can I use existing controlled vocabularies for DC Subject metadata?

One can assign a metadata value selected from a controlled vocabulary as the value of the element, and then qualify that element with the name or identifier of the scheme from which it is selected (the specifics of the encoding depends on the syntax being employed. Refer to DCMI specifications for details of encoding DC metadta in HTML, XML, or RDF/XML.

DCMI registers controlled vocabularies and encoding schemes to promote their use and to facilitate consistent identification within DC metadata. Application designers should review registered controlled vocabularies to determine if there is a suitable one for their application, and use the registered name of that vocabulary in their application. For example, "DDC" is the registered Name for the Dewey Decimal Classification, and should be used as the value of the qualifier. By using registered Names or tokens to designate schemes, metadata from different applications that use common controlled vocabularies are more likely to be interoperable.

The mechanics of selecting a value from such a vocabulary is dependent on the application. It is expected that metadata editors for some domains will have tools to support such selection, but it can also be done with conventional print-based references.

Can I use controlled vocabularies that are not approved by DCMI?

Yes. DCMI registers only those controlled vocabularies that have been brought to our attention. There are, of course, many others that are equally legitimate, and it has always been our intent that communities of expertise be able to leverage the value of such existing schemes in their metadata. To promote interoperability, it is recommended that application designers review registered controlled vocabularies for one that may be suitable for their application. If a controlled vocabulary of choice is not registered with DCMI, it is possible (but not mandatory) to register it. Registration assures that others who adopt this vocabulary use the same Name token in their metadta, thereby promoting interoperability.

It is important to note that DCMI 'registers' controlled vocabularies, rather than 'approving' them. Controlled vocabularies are generally the
result of substantial community expertise. It is not in the purview of DCMI to approve or disapprove such works, but rather help to make them visible for others who might choose to adopt them, and to prevent Name collisions by assigning unique tokens to identify them within DC metadata.

What are the names of the DCMI namespaces?

http://purl.org/dc/elements/1.1/
http://purl.org/dc/terms/
http://purl.org/dc/dcmitype/

Metadata associated with this resource: http://dublincore.org/resources/faq/index.shtml.rdf

DCMI and the DCMI Web site are hosted by OCLC Research.

Title: "Vocabulary Encoding Scheme Registration" issue
Modified: 2004-02-25 15:44, Wednesday
Maintainer: Tom Baker
Latest version: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/registration/
See also: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/
Description: Evolving summary of the VESR issue from a UB point of view. Past meeting actions are summarized in Appendix A, followed by a Bibliography.

Shepherd: Traugott Koch

SUMMARY (Tom)

In Bath, we will start by discussing first the possible implications of InfoURI on the very idea of setting up a DCMI registry. The following two documents have therefore been put into the main meeting packet:

http://www2.elsevier.co.uk/~tony/info/info.html
http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/CACHE/Weibel.InfoURI.Registry.pdf

If we conclude that InfoURI will not meet the need we have long recognized, only then should we proceed to a discussion of the steps to be undertaken and processes by which a DCMI registration service would function. The main sources for that discussion are the "Overview of Next Steps" (below) and the following two documents in the supplementary packet:

http://www.lub.lu.se/~traugott/drafts/vocab-scheme-Jan04.html
http://www.lub.lu.se/~traugott/drafts/vocab-guide6.html

OVERVIEW OF NEXT STEPS

In Seattle, the UB agreed to proceed with the fast-track system to "register" controlled vocabularies as Vocabulary Encoding Schemes (VESes)(see Appendix point A.8). This entails:

1. Finalize the Web tool
   1.1 Work with Harry to improve interface and functionality
2. Put into place the necessary documentation
   2.1 Update "Guidelines for Registration"
   2.2 DCMI namespace policy
      2.2.1 Modify policy, adding "http://purl.org/dc/schemes"
      2.2.2 Duplication of legacy URIs in the new namespace
   2.3 Process for merging of "registry" output into raw UB data
   2.4 Formulate and document a "good-neighbor" policy
   2.5 Reflect a "good neighbor policy" in documentation and schemas
   2.6 Set up and clarify use of JISCMAIL archive for audit trail
   2.7 Clarify criteria and processes for vetting proposals

3. Approve an initial set of encoding schemes
4. Manage the above tasks and represent the project to the public
    4.1 Create a one-stop Web page for the Registration project
    4.2 Clarify who will actually do what
    4.3 Assume overall project-management responsibility
5. Plan a workshop (out of UB scope per se)

NEXT STEPS IN DETAIL

1. Finalize the Web tool

   1.1 TASK (Traugott): As of 2004-01-04, Traugott has updated
       his summary of development work needed on the Web-based
       registration tool and will follow up with Harry
       and Stu to (hopefully) complete in January-February.
       See http://www.lub.lu.se/~traugott/drafts/vocab-scheme-Jan04.html
       and http://wip.dublincore.org/schemes/index.html.
       (See also Appendix points A.6 and A.14.)

2. Put into place the necessary documentation

   2.1 "Guidelines for registration of Vocabulary Encoding Schemes"

       TASK (Traugott): As of 2004-01-10, Traugott
       has updated [GUIDELINES] and will post
       a draft to DC-USAGE for comment. See
       http://www.lub.lu.se/~traugott/drafts/vocab-guide6.html.

       NOTE: We should look closely at the guidelines
       it provides on forming and proposing a "Name" for
       a vocabulary -- for example, appending a language
       suffix such as "-fr", etcetera. I believe there are
       some unresolved issues here with regard to the use
       of "date-stamped" URIs or of reflecting the numbers
       of specific versions of vocabularies. After talking
       with Traugott, I believe this was the intention behind
       the Seattle Action Item 13 for Traugott to draft "a
       document setting out guidelines for the creation of
       URIs for encoding schemes" (see A.12 below).

   2.2 DCMI Namespace Policy

      According to the policy as it currently stands, all new
      Encoding Schemes go into the http://purl.org/dc/terms/.
      However, now that we expect the creation of many new
      encoding schemes rather quickly and according to a
      new fast-track procedure, we need a new namespace for
      encoding schemes (i.e., http://purl.org/dc/schemes/).
      This change and addition to the namespace policy is on
      the critical path -- not just for going into production
      to register encoding schemes but even before we can
      finalize the related texts and guidelines.

This breaks down into the following tasks:

2.2.1 TASK (who?): Modify the Namespace Policy [DCMI-NAMESPACE] as follows:

-- a name for the new namespace;
-- clarify whether any aspect of the existing policy needs to be modified with respect to encoding schemes that are approved according to the fast-track procedure;
-- shepherd the draft through online DC-USAGE discussion;
-- liaise with Makx for Directorate approval;
-- present the results (hopefully completed?) in Bath.

2.2.2 TASK (who?): Duplication of legacy URIs under "http://purl.org/dc/terms" in the new namespace "http://purl.org/dc/schemes". This decision (and its implications) needs to be implemented and documented:

-- gather from the Directorate, Tom, Roland, and mailing lists any existing documentation of past discussions and edit them into a one-page clarification;
-- include documentation of how the equivalence would be declared in the formal RDF term declarations;
-- liaise with Tom (for the raw UB data in XML) and Harry (for the generated RDF schemas and Web pages) about declaring and documenting the equivalence;
-- after consulting with the Directorate on process, shepherd the one-pager through list discussion, approval, and posting on the Web, ideally in time for the Bath meeting.

2.3 TASK (Tom): Merging of "registry" output into raw UB data (see also A.7).

According to our current model, information about VESes will be recorded in two places:

-- in the back-end database to the Web tool.
This database will include administrative information -- e.g., who submitted a proposal and when.
The database will periodically output a listing of new VESes in a form that can _automatically_ be merged into the raw UB data (currently in XML).

NOTE: This functionality is on the critical path to using the Web tool -- if we cannot merge data from the Web tool into the system of XSLT scripts currently used to generate updated Web pages and RDF schemas of the DCMI terms automatically, I do not currently see a way to manage updates to our documentation with a reasonable and sustainable

level of effort. If we cannot automate the workflow from registration through to final publication so as to sustain a reasonable and efficient throughput, we should not really embark on this adventure to begin with.

-- in the formal RDF term declarations and related Web pages generated from the raw UB data (which, in turn, is automatically generated from the database above).

This entails the following:

-- liaison with Traugott to clarify whether any attributes need to be exported beyond those already used to describe existing encoding schemes;
-- decide in discussion on DC-USAGE or in Bath at what frequency terms documents should be updated to show new VESes (this entails liaison with maintainers of the DCMI Registry about the availability of new terms in the registry database);
-- liaise with Harry and the Web Team to clarify how often and by what workflow descriptions of VESes will be exported from the Web-tool database and incorporated into the raw UB term data;
-- verify that the workflow is completed and functions as intended for generating updated term documentation;
-- update the "schema" of attributes used to describe encoding schemes.

2.4 TASK (who???): formulate a policy for pointing to non-DCMI URIs created for vocabularies to which DCMI URIs have already been assigned (sometimes called a "good-neighbor" policy) (see also A.11). This entails the following:

-- describing a "DCMI philosophy" (or etiquette) for pointing to non-DCMI URIs (e.g., do we "recommend" one over the other or simply point?);
-- clarifying exactly where the non-DCMI URI will be recorded, and how that URI will be reflected in the DCMI Registry and exported for merging into the raw UB data used for generating RDF schemas and Web pages (see also 2.3).

2.5 TASK (who???): clarify and document exactly how the "good neighbor policy" (2.4) will be reflected in the RDF schema and in the terms Web pages. (See also A.4 and A.5). This entails:

-- clarifying with Roland -- who I believe had a solution for this that was discussed and for which notes exist somewhere... -- exactly how the cross-reference would

be expressed in RDF;
-- clarifying with Tom exactly what relevant field or fields will be automatically exported as a basis for generating Web pages and RDF schemas;
-- clarifying with Tom exactly how that additional information should appear in the Web documents;
-- liaising with Harry to ensure that the additional RDF assertions will be generated from the raw UB data.

2.6 TASK (Traugott): set up a JISCMAIL list to use as an archive of (all?) actions taken in the fast-track procedure (see also A.3 below). This entails:

-- clarifying exactly who needs to do what to verify and evaluate a submission, what follow-up actions they need to take, what needs to be documented and where;
-- providing a user-friendly list of actions and responsibilities for inclusion in the DCMI Usage Board process [UB-PROCESS] or in another appropriate document.

2.7 TASK (Diane and Stuart?): Documentation of process for reviewers of fast-track proposals. This should expand on Section 5 of [UB-PROCESS] clarifying exactly who is expected to do what in order to reach a fast-track decision about a proposed VES. This could take a list of such actions and responsibilities from 2.6. (See also point A.1 below.)

3. Approve an initial set of encoding schemes

When all of the above is in place, we somehow need to move this forward to the actual creation of VESes, especially for the initial set of known important vocabularies in the pipeline (see also point A.10).

Note that the restriction that proposals be accepted only from the owners and maintainers of vocabularies is slightly at odds with the notion that we would begin with "known important vocabularies" that have long been in the pipeline. In other words, we should decide whether to go ahead with the registration of some vocabularies on our own initiative, and if so, who will take that initiative and to what extent will those volunteers be expected to obtain permission from the owner/maintainers of the vocabularies in question.

Deciding how to proceed on this would be the responsibility of the overall project manager for Vocabulary Registration, in consultation with the Usage Board.

4. Manage the above tasks and representing the project to the public

4.1 TASK (Traugott?): Create a Web page describing DCMI's

project for registering VESes along with a one-stop annotated set of pointers to all relevant resources (such as [GUIDELINES], [UB-PROCESS], [NAMESPACE POLICY], DCMI terms documentation, and the DCMI Registry. This entails:

-- taking as input from 2.4 a "good-neighbor policy";
-- explaining overall philosophy, policy, and intentions (perhaps this should be where we explain that in the first instance, registration will be on the initiative of scheme owners -- i.e., the maintainer of the vocabulary in question does the registering by proposing an acronym for use in a DCMI-maintained URI and optionally supplying an owner-maintained URI for the same;
-- creating short versions of the above for posting as announcements to DC-GENERAL.

Such explanatory text could be folded into the start page for the Web tool [WEB-TOOL], which currently provides user guidance on using the tool, which would require coordination with Harry on editing a single Web page. Or the information could be split out into a separate document -- whatever seems friendliest for users. Either way, we should ensure that all Web pages of relevance to the project of registering encoding schemes be fully cross-referenced with all of the other relevant Web pages.

4.2 TASK (Traugott?): Clarify who will actually do what. This task involves defining what needs to be done (see also 2.7), but also who is going to actually do it and the process for managing the people who are doing it. Note that there has been considerable discussion between Ithaca and Seattle of a "reasonable" role for Usage Board members in processing proposals. However, experience in last year's trial run and with AskDCMI this year strongly suggest that asking UB members to claim proposals to vet will be problematic.

4.3 TASK (Traugott): In Seattle, Traugott volunteered to assume overall responsibility for the Registration issue. From my point of view, I see this as involving, among other things:

-- coordinating and motivating the people who will check, evaluate, and approve the proposals for VESes;
-- making announcements to DC-GENERAL;
-- coordinating with Tom about responsibilities to avoid unnecessary redundancies in tracking issues.

5. Plan a workshop

Note: As of December 2003, Tom, Traugott, Stu Weibel, Diane, and Stuart Sutton are discussing the possibility of holding a workshop to coordinate better between DCMI and other communities interested in the general problem of identifying and citing controlled vocabularies. While it is related to the process of Vocabulary Encoding Scheme Registration discussed here, the workshop issue is not further covered below. Note that this venue would be the appropriate place to discuss a possible use of IETF's InfoURI (Appendix point A.9).

APPENDIX A: Recent decisions and action items

2003-06-17: Ithaca meeting
2003-09-28: Seattle meeting

A.1 ITHACA ACTION ITEM (Diane and Stuart): Make necessary updates to the UB Process document.

A.2 ITHACA ACTION ITEM (Tom): Ask Directorate to advise UB on their position in regard to the legal issues surrounding encoding scheme registration. [Note: this has been done, and the opinion is that we can go ahead as long as we articulate our policies clearly.]

A.3 ITHACA ACTION ITEM (Traugott): In the interest of maintaining an audit trail all emails between UB/DCMI and the scheme owner/maintainer to be sent to a closed Jiscmail DC list for permanent retention; Traugott to ask Paul Miller.

A.4 ITHACA DECISION (according to Traugott): DCMI assigns a URI and lists URIs created by vocabulary owners. If this is the case, a "same as" relationship between the two is declared.

A.5 ITHACA ACTION ITEM (Tom and Diane): Draft new document that explains such things as 'good neighbour' policy, what the process involves, the aims of the registration service, registration help, etc.

A.6 ITHACA ACTION ITEM (Traugott): List of priorities for enhancements/changes to the scheme registration tool to be submitted to Makx.

A.7 ITHACA ACTION ITEM (Tom): Document the XML output formats that are wanted from the registration tool.

A.8 SEATTLE DECISION: UB agrees that it must proceed with Vocabulary Encoding Scheme registration.

A.9 SEATTLE DECISION: UB will consider adopting IETF's InfoURI if and when this is finalized.

A.10 SEATTLE DECISION: The UB is to aim for implementation of the registry by January or February 2004.

A.11 SEATTLE DECISION: Where schemes have existing URIs such schemes should be registered at the request of implementers.

A.12 SEATTLE ACTION ITEM 13 (Traugott): Draft a document setting out guidelines for the creation of URIs for encoding schemes.

A.13 SEATTLE ACTION ITEM 14 (Tom): Tom to remind Stu to seek advice from OCLC lawyers regarding legal issues surrounding encoding scheme registration.

A.14 SEATTLE ACTION ITEM 15 (Traugott): Questions to be posed to Harry, as part of request for updates and changes, regarding authentication and whether existing authentication facility is robust enough.

BIBLIOGRAPHY

[GUIDELINES] Guidelines for Vocabulary and Encoding Scheme Qualifiers, http://dublincore.org/usage/documents/vocabulary-guidelines/

[NAMESPACE-POLICY] DCMI Namespace Policy, http://dublincore.org/documents/dcmi-namespace/.

[DCMI-PRINCIPLES] DCMI Grammatical Principles, http://dublincore.org/usage/documents/principles/.

[UB-PROCESS] DCMI Usage Board Process, http://dublincore.org/usage/documents/process/

[WEB-TOOL] Vocabulary Scheme Registration [a Web-based tool], http://wip.dublincore.org/schemes/index.html.

About "*info*" URIs

Frequently Asked Questions

This version dated January 5, 2004.

(The latest version of this FAQ is always available from the ***info*** website: <http://info-uri.info/>.)

Feedback on this FAQ should be directed to <mailto:t.hammond@elsevier.com>.

Disclaimer: This FAQ is provided for information purposes only and should be treated as a work in progress.

The Questions

A. **Background (I)** <<

B. **Background (II)** <<

C. **The *info* URI and Other URI Schemes** <<

D. **The *info* Registry** <<

E. **Syntax Considerations** <<

The Answers

A. Background (I)

1. **What is a URI scheme?** <<

   A Uniform Resource Identifier (URI) provides a simple and extensible means for identifying a resource within the Web global information architecture. Each URI begins with a scheme name that refers to a specification for assigning identifiers within that scheme. As such, the URI syntax is a federated and extensible naming system wherein each scheme's specification may further restrict the syntax and semantics of identifiers using that scheme.

2. **Why was it necessary to develop the *info* URI scheme?** <<

   The ***info*** URI scheme is being developed from within the library and publishing communities to expedite the referencing by URIs of information assets that have identifiers in public namespaces but have no representation within the URI allocation.

   For various reasons (both cultural and technical) the creation and registration of a new URI scheme or URN namespace to support a given public namespace under the URI allocation may not have been attempted by the authority for that namespace. It is precisely to facilitate the representation of these public namespaces within the URI allocation that the ***info*** URI scheme has been developed.

3. **What is the motivation behind the *info* URI scheme?** <<

   The motivation behind developing the ***info*** URI scheme is to allow legacy identification systems to become part of the Web global information architecture so that the information assets they identify can be referenced by Web-based description technologies such as XLink, RDF or Topic Maps. Note that we are concerned with "information assets", not "digital assets" *per se* - the information assets may be variously digital, physical or conceptual.

4. **Does the *info* URI scheme compete with other URI schemes?** <<

   No. The ***info*** URI scheme does not compete with independent URI registrations but rather cooperates with independent URI registrations by providing a lightweight early URI registration mechanism to support referencing of public information assets ahead of any possible subsequent URI scheme or URN namespace application. Note that in the majority of cases no subsequent URI scheme or URN namespace application will be made by a Namespace Authority as the ***info*** resource identifier alone will be sufficient in providing an identification service. Only if additional services are required would a Namespace Authority seek to register independently.

5. **Will the *info* URI scheme break Web applications?** <<

   No. ***info*** URIs support no operations beyond representation of identity and comparison with other ***info*** URIs, and therefore any Web applications that can recognize a URI within a URI usage context should not break with this additional URI scheme. ***info*** URIs are conformant to the general syntax requirements of URIs as laid out in RFC 2396 (and as revised in RFC 2396bis) and as such are readily parseable and recognizable as legitimate URI strings.

6. **Where is the latest specification for *info*?** <<

   The latest specification for ***info*** is the published Internet-Draft: "draft-vandesompel-info-uri-01.txt". (This revision supercedes the earlier draft "draft-vandesompel-info-uri-00.txt".)

7. **Who is developing the *info* URI scheme?** <<

   The ***info*** URI scheme has been developed by members of the library and publishing communities working together under the auspices of ANSI/NISO.

   Representing the library communities are Herbert Van de Sompel of LANL (Los Alamos National Laboratory) Research Library and Stu Weibel of OCLC (Online Computer Library Center), while representing the publishing communities are Tony Hammond of Elsevier and Eamonn Neylon of Manifest Solutions, a publishing technology consultancy.

   NISO, the National Information Standards Organization, is a standards developing organization accredited to the International Organization for Standards (ISO) by the American National Standards Institute (ANSI), the sole US representative to ISO.

8. **Why is it called *info*?** <<

   It is called ***info*** as a shorthand for the information asset that it references by means of a URI.

Another reading of ***info*** is that ***info*** exists solely to provide information about an information asset, the information disclosed being restricted to identity alone. Richer sets of information, such as authority metadata, would require resolution services which are not supported by ***info*** and hence would require a corresponding independent URI scheme or URN namespace application.

9. **What is the current status of the *info* URI scheme?** <<

   The Internet-Draft "draft-vandesompel-info-uri-00.txt" for the ***info*** URI scheme was first announced September 30th, 2003. A revision to this draft "draft-vandesompel-info-uri-01.txt" was subsequently announced December 6th, 2003.

   The ***info*** URI scheme has already been fully embraced by one Web description technology: the OpenURL Framework.

B. Background (II)

1. **What is the value proposition of the *info* URI scheme?** <<

   The value proposition of ***info*** URIs is to facilitate Namespace Authorities in representing identifiers from their namespaces in URI form and so to make those identifiers available for public use in Web-based description technologies independently of any Web infrastructure considerations on behalf of the Namespace Authorities.

   As such, ***info*** URIs are designed to be simple to deploy, both in terms of namespace registration and in the creation and general usage of these identifiers. These simplifications are fully in line with the ***info*** URI remit of providing a lightweight URI registration mechanism to facilitate the referencing of information assets under the URI allocation.

2. **What is unique about the *info* URI scheme?** <<

   The unique aspect of ***info*** URIs is that ***info*** URIs are not dereferenceable. The sole purpose of ***info*** is the disclosure of the identity of an information asset from a public namespace.

3. **And just what is a non-dereferenceable URI?** <<

   A non-dereferenceable URI is one that has no representations associated with the resource that it identifies, and as such no retrieval actions can be performed. In short, a user "click" event on such a URI (e.g. a user clicking on a hyperlink in a browser) would yield no result.

4. **So, why are *info* URIs non-dereferenceable?** <<

   info is focused exclusively on supporting identity. As such, this restriction on the functionality of ***info*** URIs to identity alone leads to the following considerations:

   - Dramatically simplifies ***info*** operational behaviours, and consequently the operational expectations of an ***info*** URI, which are limited to URI string presentation and comparison.
   - Avoids overhead in new ***info*** namespace registrations by rejecting any possibility of

resolution mechanisms.
(Note that public namespaces may have functionality associated with them which is accessible on the Internet, but ***info*** does not provide an access route into this functionality. Applications wishing to tap into this functionality must resort to out-of-band mechanisms.)

- Avoids overhead in the management of a resolution infrastructure and maintenance of resolution targets for ***info*** URIs.
- Requires that a Namespace Authority make an independent URI scheme or URN namespace application if services or other functionalities are required (e.g. authority metadata).
- Does not run foul of the mantra "*many URI schemes considered harmful*" as this is primarily concerned with the need for applications to support resolution mechanisms.

Note that this stance on non-dereference is in contrast to other URI schemes which are generally dereferenceable (or otherwise actionable by applications), even if not all examples within a particular scheme are actually dereferenceable. This holds true for both location-dependent URIs (e.g. HTTP URIs) as well as for location-independent URIs (e.g. URN URIs).

5. **Why identify information assets with URIs anyway?** <<

The real need for presenting legacy identifiers in URI form is that many Web-based description technologies (e.g. XLink, RDF, Topic Maps, OpenURL) recognize URIs as the only form of globally unique identifier.

Besides this, the URI naming architecture offers to identifiers a common base syntax, a common base semantics, and a common nomenclature for talking about identifier constructs (e.g. URIs reference "resources" while generally identifiers for information assets will reference diverse kinds of things - objects, records, etc.). The URI thus provides a uniform (and unifying) naming architecture for identifiers.

The examples FAQ included here might afford some insight into the desirability of rendering identifiers as URIs, cf. the identifiers in their native form and as rendered in URI form.

6. **Can anyone use their namespaces under *info*?** <<

No. Namespaces used under ***info*** must be registered in the ***info*** Registry.

7. **Who is entitled to register under *info*?** <<

A Namespace Authority is entitled to register a recognized namespace after suitable review.

A Namespace Authority is the body that owns and manages a public namespace.

8. **Which namespaces are eligible for registration under *info*?** <<

The namespaces eligible for registration under ***info*** will typically be those of interest to the publishing, library and media communities. These communities necessarily have a very wide purview. Candidate namespaces will be those that are for public use only and that are not

part of the URI allocation. Non-public namespaces are not eligible for registration.

For all registered namespaces the reader is advised to consult the ***info*** Registry. The following registered namespaces are typical:

- **bibcodes** - NASA Astrophysics Data System Bibcodes (~3.5m)
- **doi** - International DOI Foundation Digital Object Identifiers (~11m)
- **lccn** - Library of Congress Control Numbers (~12m)
- **oclcnum** - Online Computer Library Center (OCLC) Numbers (~52m)
- **ofi** - NISO OpenURL Framework Identifiers (~40)
- **pmid** - National Library of Medicine PubMed Identifiers (~10m)
- **sid** - Source Identifiers (as used by the OpenURL Framework)

(Note: To put those numbers into perspective, as of this writing the accumulated totals represent some 2.6% of Google space.)

9. **Does registration under *info* limit any future URI registration?** <<

   No. On the contrary, by occupying its own toplevel URI scheme, ***info*** does not restrict a Namespace Authority from proceeding with registration under any portion of the URI allocation either as a toplevel URI scheme or as a URN namespace. In particular, an independent ***info*** URI scheme does not interfere with any of the existing URI namespaces which have their own defined semantics.

C. Comparison with Other URI Schemes

1. **What considerations are there regarding URIs?** <<

   Some general considerations regarding URIs are:

   - *Generality* - URI schemes are divided between generality and speciality as regards their reliance upon an Internet transport protocol.
   - *Location dependence* - URI schemes are variously categorized as location-dependent or location-independent with respect to whether the URI contains a network authority component.
   - *Character set* - The set of allowable characters within a URI is a subset of ASCII. Certain ASCII characters are excluded, while others are reserved within various contexts within the URI string according to the URI scheme in question. Excluded and reserved characters must be %-escaped as described in RFC 2396.

     The wider the character set that a given URI scheme supports, the more easily it can function as a substrate for legacy identifiers.

2. **Why not just use HTTP URIs?** <<

   HTTP URIs are Internet protocol elements for referencing hypertext documents which can be

retrieved from a network authority using the HTTP transfer protocol. There is a common expectation that HTTP URIs can be dereferenced.

The following considerations hold in respect of HTTP URIs:

- HTTP URIs are inappropriate for ***info*** namespaces because HTTP URIs provide:
  - network transport
  - network path (discovery obstacle)
  - strong dereference expectation
  - poor branding (network path overhead)

- A transport mechanism adds meaningless semantic overhead to nondereferencable URIs.
- Absolute HTTP URIs include a network path (comprised of an authority component and a hierarchical path component). ***info*** namespaces may not have (or may not make) any network authority available. A central network authority would also be inappropriate as this would introduce a dependency between a third party namespace and a central network authority.

  Further, were ***info*** namespaces to make a network authority available they would each have to publish the particular hierarchical path syntax employed by that network authority. A central network authority would mitigate this requirement by providing a single path syntax, although it would still need to publish that path syntax.

- Use of HTTP URIs might only encourage the provisioning of resource representations (e.g. metadata descriptions) which could conflict with representations provided under any possible future URI registration on the part of the Namespace Authority. Further, if HTTP URIs were used to provide resource representations, it must be recognized that managing the namespace and infrastructure is a costly enterprise that may not be appropriate or cost effective in a given business context.
- The network path of HTTP URIs adds unnecessary string overhead and consequent loss of branding for legacy identifiers.

3. Well then, why not just use URN URIs? <<

URN URIs are Internet protocol elements for referencing resources using persistent and location-independent identifiers, representations of which may be retrieved using various resolution mechanisms. There is a common expectation that URN URIs can be dereferenced, once suitable resolution mechanisms are defined (e.g. DDDS or other proprietary mechanisms). Indeed, RFC 1737 goes so far as to make a strong recommendation that "*there be a mapping between the names generated by each naming authority and URLs*".

Use of URN URIs requires a URN namespace registration. An informal URN namespace is of limited utility because its numerical nature obliterates any branding or name recognition and effectively renders the namespace anonymous. A formal URN namespace, on the other hand, would require a more substantial review than a corresponding registration under the ***info*** Registry. Based on experience with the initial ***info*** namespace target group, it is unlikely that

many Namespace Authorities will proceed with independent applications as the burden of registering a URN namespace is high, especially in the case of organizations that are not strongly steeped in technology.

One particular impediment in applying for a URN namespace for ***info*** is that this would compromise any possible future URN namespace registration that a Namespace Authority might seek to make in respect of considerations of persistence, location independence and/or dereference to resource representations.

The following considerations hold in respect of URN URIs:

- URN URIs are inappropriate for ***info*** namespaces because URN URIs provide:
  - claims of persistence of resource identifiers
  - dereference expectation
  - no delegated naming responsibility
  - restricted syntax (no hierarchy)
  - no support for fragment identifiers
  - poor branding and extra semantic layer (additional namespace tier)

- ***info*** URIs make no claims on persistence. ***info*** URIs may be location independent and in consequence may enjoy some degree of persistence, but ***info*** does not make these assertions. Instead ***info*** is neutral with respect to identifier persistence.
- Use of URN URIs might only encourage the provisioning of resource representations (e.g. metadata descriptions) which could conflict with representations provided under any possible future URI registration on the part of the Namespace Authority. Further, if URN URIs were used to provide resource representations, it must be recognized that managing the namespace and infrastructure is a costly enterprise that may not be appropriate or cost effective in a given business context.
- For ***info*** to operate as a URN namespace would require that ***info*** be constituted as a delegated naming authority. It is not clear that a URN namespace would be an appropriate choice for such naming authority delegation.
- Syntactically, URN URIs do not support hierarchy (in URI syntax hierarchy proceeds through the "/" character) and are thus more difficult to use with legacy identifiers because of their restricted character set. Other characters reserved by URN URIs, but allowed by ***info*** URIs are "&" and "~".

  For a demonstration in the difficulty of mapping legacy identifiers the reader is referred to RFC 3151 which provides a set of complex transcriptions for mapping SGML formal public identifiers onto the URN URI syntax. Formal public identifiers would have been more readily presented under the more expressive ***info*** syntax.

- Additionally, URN URIs do not support fragment identifiers thus not allowing the identification of secondary resources with respect to a primary resource. This is a pratical requirement that ***info*** supports.

- With ***info*** as a URN namespace, the ***info*** namespaces would then become sub-sub-namespaces, with a consequent loss of branding. This would also introduce three tiers of semantic layers for an implementation to navigate.

4. **Are there any other URI schemes that might be considered?** <<

   Two other generic URI schemes that might be considered are DATA URIs and TAG URIs.

5. **So, why not use DATA URIs?** <<

   DATA URIs are Internet protocol elements for the referencing of inline resources, i.e. the reference is an immediate address.

   The following considerations hold in respect of DATA URIs:

   - DATA URIs are inappropriate for ***info*** namespaces because DATA URIs provide:
     - no authority mechanism (network authority or registry)
     - no globally unique identifiers

   - The DATA URI provides a data resource inline without reference to any network authority or registry. There is thus no record for the namespaces that would need to be supported.
   - Without an authority there can be no guarantee that DATA URIs are globally unique. This guarantee of uniqueness is the prime motivator for the ***info*** URI scheme.

6. **Or else, why not use TAG URIs?** <<

   TAG URIs are (proposed) Internet protocol elements for referencing resources that are globally unique across network space and time. (See the last Internet-Draft, now expired, for further details.) The basic premise is that a specific identifier is prefixed with a TAG authority which is constructed from a fully qualified domain name (either used alone or embedded within an email address) and a date in ISO 8601 format. The TAG authority should thus guarantee that the TAG URI is globally unique. TAG URIs may or may not have resolution mechanisms, although those mechanisms are not specified.

   The following considerations hold in respect of TAG URIs:

   - TAG URIs are inappropriate for ***info*** namespaces because TAG URIs provide:
     - network authority (discovery obstacle)
     - redundant date component (requires adoption of a date convention)
     - dereference expectation
     - poor branding (string overhead from network authority and date components)
     - unclear semantics (URI and/or URN namespace?)
     - no IANA registration

   - TAG URIs include a network authority component. ***info*** namespaces may not have (or may

not make) any network authority available. A central network authority would also be inappropriate as this would introduce a dependency between a third party namespace and a central network authority. Further, the form of disclosing the network authority component (either as an independent string or embedded within an email address) would need to be published.

- Another difficulty is that the TAG URI is designed for minting new identifiers whereas ***info*** is purely concerned with proxying legacy identifiers (i.e. already minted) onto the URI namespace. Since there is a date component in the TAG URI the only means of making this work as a basis for supporting legacy identifiers would be to adopt some convention whereby a known date would be used for all the identifiers (which obviously obviates the need for a date component).
- Use of TAG URIs might only encourage the provisioning of resource representations (e.g. metadata descriptions) which could conflict with representations provided under any possible future URI registration on the part of the Namespace Authority. Further, if TAG URIs were used to provide resource representations, it must be recognized that managing the namespace and infrastructure is a costly enterprise that may not be appropriate or cost effective in a given business context.
- The TAG authority of TAG URIs adds unnecessary string overhead and consequent loss of branding for legacy identifiers.
- It is unclear whether TAG URIs should be regarded as location-dependent or location-independent. The Internet-Draft makes a dual application for TAG as a toplevel URI scheme, as well as a URN namespace.
- At the current time the TAG URI application has not been approved for publication under the IANA registry. The Internet-Draft is now expired.

D. The *info* Registry

1. **What is the *info* Registry?** < <

   The ***info*** Registry provides a mechanism for the registration of public namespaces that are used for the identification of information assets, and that are not referenceable within the URI allocation.

2. **Where is the *info* Registry located?** < <

   The ***info*** Registry is located on the ***info*** website at: <http://info-uri.info/registry/>

3. **Who maintains and operates the *info* Registry?** < <

   NISO is the Maintenance Agency for the ***info*** Registry and may delegate operational responsibility to an operating body, or Registry Operator.

4. **What policies govern the *info* Registry?** < <

   A publicly articulated policy established under NISO governance will be made available on the ***info*** website <http://info-uri.info/>. The ***info*** Registry policy defines a review process for

candidate namespaces and provides measures of quality control and suitability for entry of namespaces

5. **What is the registration procedure for an *info* namespace?** <<

   The ***info*** Registry provides an online registration form which solicits the relevant fields. This information will then be made available to a public approval process. Subject to a favourable review the namespace will be added to the ***info*** Registry.

6. **What functionalities does the *info* Registry present?** <<

   The ***info*** Registry is publicly accessible and supports discovery (by both humans and machines) of:

   - String literals identifying the namespaces.
   - Names and contact information of Namespace Authorities.
   - Syntax requirement for identifiers maintained in such namespaces.
   - Normalization methodology for identifiers maintained in such namespaces.
   - Ancillary documentation.

   Disclosure of Registry records will be both in a human presentational form as well as in structured markup form for machine automation. Currently the Registry records are presented as XHTML documents but are also available as "oai_dc" XML records and are harvestable using the OAI-PMH protocol. Other presentations, such as RSS, may subsequently be made available.

E. Syntax Considerations

1. **What are the syntax highlights for an *info* URI?** <<

   The ***info*** URI syntax is very straightforward:

   ```
   "info:" namespace "/" identifier [ "#" fragment ]
   ```

   where `namespace` is a registered namespace token, `identifier` is a %-escaped identifier, and `fragment` is an optional %-escaped identifier to a secondary resource.

   The ***info*** URI syntax introduces certain restrictions on the generic URI syntax:

   - No network authority component allowed.
   - No query component allowed.
   - No absolute path allowed (although relative paths are admitted).
   - No relative URI references allowed.

2. **What is the full syntax for an *info* URI?** <<

The ***info*** URI syntax is generally conformant with the generic URI syntax defined in RFC 2396. The only point of departure is to make use of the new "segment" production as generalized in RFC 2396bis the revision to RFC 2396.

This specification uses the Augmented Backus-Naur Form (ABNF) notation of RFC 2234 to define the URI. The following core ABNF productions are used by this specification as defined by Section 6.1 of RFC 2234: ALPHA, DIGIT, HEXDIG.

The ***info*** URI syntax is presented in two parts. Part A contains productions specific to the ***info*** URI scheme:

```
info-URI        = info-scheme ":" info-identifier [ "#" fragment ]

info-scheme     = "info"

info-identifier = namespace "/" identifier

namespace       = scheme

identifier      = path-segments
```

Part B contains generic productions from the RFC 2396 revision RFC 2396bis, which are repeated here both for completeness and for reference.

```
scheme          = ALPHA *( ALPHA / DIGIT / "+" / "-" / "." )

path-segments   = segment *( "/" segment )

segment         = *pchar

fragment        = *( pchar / "/" / "?" )

pchar           = unreserved / escaped / ";" /
                  ":" / "@" / "&" / "=" / "+" / "$" / ","

unreserved      = ALPHA / DIGIT / mark

escaped         = "%" HEXDIG HEXDIG

mark            = "-" / "_" / "." / "!" / "~" / "*" / "'" /
```

```
                    "(" / ")"
```

3. **What does the fragment signify for an *info* URI?** <<

   The fragment on an ***info*** URI signifies a secondary resource with respect to the primary resource identified by an absolute ***info*** URI without a fragment.

F. Examples of Usage

1. **What do *info* URIs look like?** <<

   Some examples of (syntactically) valid ***info*** URIs are given below together with the identifiers they reference in their native form and the information assets that they identify:

   a. **Dewey Decimal Classification**

   b. **Library of Congress Control Number**

   c. **Serial Item and Contribution Identifier (SICI)**

   d. **Astrophysics Data System Bibcode**

   e. **PubMed Identifier**

a. Dewey Decimal Classification <<

Notes: **ddc** is the ***info*** namespace component for a Dewey Decimal Classification namespace and **22/eng//004.678** is the identifier component for an identifier of an information asset within that namespace.

```
info:ddc/22/eng//004.678

22/eng//004.678

Information Asset: Vocabulary Term

  "Internet"
```

b. Library of Congress Control Number <<

Notes: **lccn** is the ***info*** namespace component for a Library of Congress Control Number namespace and **2002022641** is the identifier component for an identifier of an information asset within that namespace.

info:lccn/2002022641

2002022641

Information Asset: Book

"Newcomer, Eric. Understanding Web services: XML, WSDL, SOAP, and UDDI. Boston: Addison-Wesley, 2002."

c. Serial Item and Contribution Identifier (SICI) <<

Notes: **sici** is the ***info*** namespace component for a Serial Item and Contribution Identifier namespace and **0363-0277(19950315)120:5%3C%3E1.0.TX;2-V** is the identifier component for an identifier of an information asset in that namespace in escaped form, or in unescaped form **0363-0277(19950315)120:5<>1.0.TX;2-V**.

info:sici/0363-0277(19950315)120:5%3C%3E1.0.TX;2-V

0363-0277(19950315)120:5<>1.0.TX;2-V

Information Asset: Journal Issue

"Library Journal, Vol. 120, no. 5. March 15, 1995."

d. Astrophysics Data System Bibcode <<

Notes: **bibcode** is the ***info*** namespace component for an Astrophysics Data System bibcode namespace and **2003Icar..163..263Z** is the identifier component for an identifier of an information asset within that namespace.

info:bibcode/2003Icar..163..263Z

2003Icar..163..263Z

Information Asset: Abstract of Journal Article

"K. Zahnle, P. Schenk, H. Levison and L. Dones, Cratering rates in the outer Solar System, Icarus, 163 (2003) pp. 263-289."

e. PubMed Identifier <<

Notes: **pmid** is the ***info*** namespace component for a PubMed Identifier namespace and **12376099** is the identifier component for an identifier of an information asset within that namespace.

info:pmid/12376099

12376099

Information Asset: Abstract of Journal Article

"Wijesuriya SD, Bristow J, Miller WL. Localization and analysis of the principal promoter for human tenascin-X. Genomics. 2002 Oct;80(4):443-52."

2. **How are *info* URIs used in Web applications?** <<

Some examples of (syntactically) valid ***info*** URIs are given below as they would be used by applications:

a. **RDF Graph**

b. **Topic Map**

c. **Extended XLink**

d. **RSS 1.0 Feed - ContextObject (1)**

e. **OpenURL Link - ContextObject (2)**

a. RDF Graph < <

Notes: This RDF graph asserts a set of statements about the resource referenced by the URI **info:pii/S0888-7543(02)96852-7** (which identifies the Publisher Item Identifier S0888-7543(02)96852-7) using the Dublin Core vocabulary.

```
<rdf:Description about="info:pii/S0888-7543(02)96852-7">
  <dc:creator>Wijesuriya, S.D.</dc:creator>
  <dc:title>Localization and analysis of the principal promoter
    for human tenascin-X.</dc:title>
  <dc:identifier>info:doi/10.1006/geno.2002.6852</dc:identifier>
</rdf:Description>
```

b. Topic Map < <

Notes: This topic map entry defines the topic internet which has a public subject indicator as referenced by the URI **info:ddc/22/eng//004.678** (which identifies the Dewey Decimal Classification Internet) and a base name string of Internet.

```
<topic id="internet">
  <subjectIdentity>
    <subjectIndicatorRef xlink:href="info:ddc/22/eng//004.678" />
  </subjectIdentity>
  <baseName id="_id123">
    <baseNameString>Internet<baseNameString>
  </baseName>
</topic>
```

c. Extended XLink <<

Notes:

```
<ce:inter-refs>
  <ce:inter-refs-text id="interref8">Parts I and II</ce:inter-refs-text>
  <ce:inter-ref-end xlink:href="info:pii/S0167-8396(00)00009-1">
    <ce:inter-ref-title>Part I</ce:inter-ref-title>
  </ce:inter-ref-end>
  <ce:inter-ref-end xlink:href="info:pii/S0167-8396(00)00010-8">
    <ce:inter-ref-title>Part II</ce:inter-ref-title>
  </ce:inter-ref-end>
  <ce:inter-refs-link/>
</ce:inter-refs>
```

d. RSS 1.0 Feed - ContextObject (1) <<

Notes: The ContextObject shown here in an RSS feed is making use of the `mod_context` RSS 1.0 module. This is the same ContextObject as shown above in Example e) but with a different serialization.

```
<channel rdf:about="http://rss.example.com/">
  <title>My conetxt-sensitive RSS feed</title>
  <link>http://rss.example.com/"</link>

  <!-- ContextObject => { -->
  <ctx:ctx_ver>Z39.88-2004</ctx:ctx_ver>
  <ctx:rft_id>
    info:sici/0363-0277(19950315)120:5%3C%3E1.0.TX;2-V
  </ctx:rft_id>
  <ctx:req_id>
    mailto:a.n.other@example.net
  </ctx:req_id>
  <!-- } -->

  <items>
    <rdf:Seq>
      <rdf:li rdf:resource="http://rss.example.com/resource/1" />
      <rdf:li rdf:resource="http://rss.example.com/resource/2" />
    </rdf:Seq>
  </items>
</channel>
```

e. OpenURL Link - ContextObject (2) <<

Notes: The ContextObject is shown here on an OpenURL which is requesting context-sensitive services from the link resolver <http://link.example.com/resolver> for the referent entity identified by the URI **info:sici/0363-0277(19950315)120:5%3C%3E1.0.TX;2-V** (which identifies the SICI code 0363-0277(19950315)120:5<>1.0.TX;2-V) and the requester entity identified by the URI mailto:a.n.other@example.net. The example OpenURL is text wrapped with whitespace for readability.

```
http://link.example.com/resolver?url_ver=Z39.88-2004
   &rft_id=info:sici/0363-0277(19950315)120:5%3C%3E1.0.TX;2-V
   &req_id=mailto:a.n.other@example.net
```

G. Comparing *info* URIs

1. **Why is comparison of URIs important?** <<

   A prime motivator for comparison of URIs is to improve performance in retrieval operations by using caching mechanisms. A secondary motivator is to equivalence resource identifiers in Web descriptions of information structures to allow those structures to be normalized or merged with other like structures, e.g. the merging of two RDF graphs or the merging of two Topic Maps.

   As far as ***info*** URIs are concerned there are no associated retrieval mechanisms, and the prime motivator for comparison of ***info*** URIs is to support the equivalencing of identifiers for information assets.

2. **What steps should be taken to compare two *info* URIs?** <<

   info URIs only exist in absolute form - no relative URI forms are allowed. The normalization steps that should be applied are with respect to the three components of the URI: the "scheme" component, the "namespace" component and the "identifier" component.

   The following generic normalization steps should be applied:

   a. Normalize the case of the "scheme" component to be lowercase.
   b. Normalize the case of the "namespace" component to be lowercase.
   c. Unescape all unreserved %-escaped characters in the "namespace" and "identifier" components.
   d. Normalize the case of any %-escaped characters in the "namespace" and "identifier" components to be uppercase.

   The subsequent namespace-specific normalization steps may be applied:

   e. Normalize the case of the "identifier" component as per any rules that may be recorded in the Registry.
   f. Normalize any punctuation characters in the "identifier" component as per any rules that may be recorded in the Registry.

3. **Is there an example of comparing *info* URIs?** <<

   Before comparing ***info*** URIs they should be normalized by applying the standard ***info*** normalization rules (lowercasing the "scheme" and "namespace" components, unescaping non-reserved characters and uppercasing any %-escaped characters that remain). In the example below four unnormalized forms (Step A), are reduced to normalized forms (Step B).

   The Registry may be consulted for namespace-specific rules on case normalization (Step C) and punctuation normalization (Step D). If these namespace-specific rules are available ***info*** URIs may be reduced to a unique canonical form.

```
   Step A - Unnormalized Forms

   U1. INFO:PII/S0888-7543(02)96852-7
   U2. info:PII/S0888754302968527
   U3. info:pii/S0888%2D7543%2802%2996852%2D7
   U4. info:pii/s0888-7543(02)96852-7

   Step B - Normalized Forms

   N1. info:pii/S0888-7543(02)96852-7
   N2. info:pii/S0888754302968527
   N3. info:pii/S0888-7543(02)96852-7
   N4. info:pii/s0888-7543(02)96852-7

   Step C - Normalized Forms - pii Case Rules Applied

N1,N3. info:pii/S0888-7543(02)96852-7
   N2. info:pii/S0888754302968527

   Step D - Normalized Forms - pii Case & Punctuation Rules Applied

   N2. info:pii/S0888754302968527
```

H. Historical Perspective

1. **What is the historical background to *info*?** <<

   The effort to create the ***info*** URI scheme emerged from the ANSI/NISO process to standardize the OpenURL Framework for Context-Sensitive Services (see the Part 1 and Part 2 ballot documents), which requires the ability to describe resources by means of globally unique identifiers. The Draft Standard for Trial Use released for Public Comment introduced a "proprietary" naming architecture which allowed information assets to be referenced by means of widely used non-URI identifiers (e.g. National Library of Medicine PubMed identifiers, International DOI Foundation Digital Object Identifiers, NASA Astrophysics Data System Bibcodes, and others) which would be registered under the OpenURL Framework.

   The Public Comment period started March 12th, 2003, when the first document was released. Early public feedback led to the decision to fundamentally revise the naming architecture, and to base all resource identification requirements within the OpenURL Framework on URIs alone. Because it was deemed unrealistic to expect that all namespaces required in the OpenURL Framework would be registered within the URI allocation by the respective Namespace Authorities, the ***info*** URI effort was launched.

2. **What consultations preceded the development of *info*?** [<<]

   Development of the ***info*** URI scheme is being conducted under the auspices of NISO, together with consultation from the IETF and the W3C. A consultation document "Bootstrapping the Web" was prepared to analyse the options available.

   On June 19th 2003, representatives from NISO, NISO Committee AX on OpenURL Framework standardization, the IETF, and the W3C met to discuss the requirements with respect to the identification of resources in the OpenURL Framework. More specifically, this group discussed how to handle the ORI and XRI Naming Environments introduced in the OpenURL Framework Draft Standard for Trial Use as an integral part of the Internet's URI environment. There was a general consensus to proceed with registration of a toplevel URI scheme and follow on meetings with representatives from the various communities were scheduled.

3. **What were the subsequent steps in the development of *info*?** [<<]

   It was recognized that these resource identifiers would have a much wider applicability to many Web-based applications and it was decided to decouple this URI scheme from the OpenURL Framework and to name it ***info***. An Internet-Draft was duly prepared and posted to the Internet-Drafts repository on September 30th, 2003. This was also widely communicated to various mailing lists and there was much discussion of the new ***info*** URI scheme on the public uri@w3.org mailing list.

4. **What are the next steps in the development of *info*?** [<<]

   A revision of the ***info*** Internet-Draft "draft-vandesompel-info-uri-01.txt" was announced December 6th, 2003 which together with this FAQ will guide Web applications, such as OpenURL Framework implementations and others, in using the ***info*** URI scheme.

   An early ***info*** Registry has also been established and namespace registrations are currently being processed. A Registry policy document is available as a work in progress from the ***info*** website.

   NISO Committee AX completed its work on the OpenURL Framework standardization at its last F2F meeting on October 27th/28th and handed over to NISO the final documents. It is expected that this draft standard will be successfully ballotted mid-Q1, 2004, and that it will be published as ANSI/NISO Z39.88-2004.

- DRAFT- "info" URI Registry Policy *- DRAFT -*

Stuart Weibel
OCLC Office of Research
2003-12-03

This document is a work in progress.
Comments and suggestions are welcome:
<mailto:weibel@oclc.org>

I. Introduction

This document outlines policies pertaining to the operation and management of the "info" URI Registry implemented in support of the "info" URI scheme described fully in [1].

The "info" URI Registry contains namespace declarations registered in support of resolution-independent identification of information assets with identifiers in public namespaces.

II. Glossary

The following terms have specific meanings in this document.

- IDENTIFIER
  A literal character string which uniquely identifies an information resource within a namespace identified by a NAMESPACE TOKEN

- INFO-SCHEME
  The *info:* token, delimiting the beginning of a complete "info" URI

- INFO-IDENTIFIER
  The combination of a NAMESPACE TOKEN and IDENTIFIER

- "info" URI
  A public identifier designed to reference information assets with identifiers in public namespaces that are not part of the resolvable URI allocation (the http: protocol). An "info" URI consists of the INFO-SCHEME token (*info:*) , and an INFO-IDENTIFIER

- "info" URI REGISTRY OPERATOR
  The organization charged with the operational responsibility for administering and operating the "info" URI Registry

- "info" URI Namespace Review List
  An open subscription mailing list designated for the purpose of reviewing proposed NAMESPACE TOKENs used in "info" URIs

- NAMESPACE AUTHORITY
  An organization that owns or manages a public namespace

- NAMESPACE TOKEN
  The registered token specifying the managed namespace in which an identifier is specified

III. Characteristics of a reserved NAMESPACE TOKEN

Syntactic characteristics of an INFO-IDENTIFIER are defined in [1].

Specification of a NAMESPACE TOKEN is at the discretion of the NAMESPACE AUTHORITY requesting the registration. Generally such tokens will reflect the identity of the organization managing the namespace, or the identity of the namespace itself.

It is recommended that registered tokens be relatively short and recognizable by the constituency for which they are intended. These tokens represent, in part, a branding of a given namespace.

IV. Who can register a NAMESPACE TOKEN

NAMESPACE TOKENs can only be registered by the NAMESPACE AUTHORITY for the identifier namespace in question.

V. Speculative NAMESPACE TOKEN Registration

Speculative or pre-emptive registration of NAMESPACE TOKENs is not permitted. Registrants attest that a proposed NAMESPACE TOKEN is currently in use. NAMESPACE TOKENs that are requested as part of a planned business activity can be reserved for 12 months, pending demonstrated public use.

VI. Registration Procedure

Registration of an “info” URI NAMESPACE TOKEN proceeds in the following phases:

- Proposal Submission

  NAMESPACE AUTHORITY submits a form-based registration proposal, including the desired NAMESPACE-TOKEN.

  “info” URI Registry Operator confirms receipt of proposal, and the status of the proposal is set to INTERNAL-REVIEW

- Internal Review

  The “info” URI Registry Operator reviews the proposed NAMESPACE TOKEN for potential collision with other registered names in the domain of Internet protocols [List of criteria to be added].

  The “info” URI Registry Operator reserves the right to reject a request for a particular NAMESPACE TOKEN on the basis of recognized name collisions or high potential for intellectual property disputes.

  In the case that grounds for rejecting the proposed NAMESPACE TOKEN are identified, the “info” URI Registry Operator so notifies the NAMESPACE AUTHORITY and negotiates a re-submission.

  In the absence of grounds for rejection of a NAMESPACE TOKEN, the “info” URI Registry Operator so notifies the NAMESPACE AUTHORITY, posts notice of the proposal on the “info” URI Namespace Review List for public comment, and sets the status of the proposal to PUBLIC-COMMENT.

- Public Comment

  In the case of substantive objection on the “info” URI Namespace Review List to a proposed NAMESPACE TOKEN, the proposal will be submitted to the “info” URI Review Board for formal review and negotiation with the NAMESPACE AUTHORITY that has proposed the token.

  Public comment on a proposed NAMESPACE TOKEN will extend for an interval determined by the “info” URI Registry Operator to allow for public comment on the proposal. In the absence of substantive objections, the “info” URI Registry Operator will establish a public Namespace Record in the Registry and set the status of the proposal to REGISTERED-ACTIVE

  A NAMESPACE TOKEN proposal will receive REGISTERED-PROVISIONAL status if it is part of a planned service, but not in active use.

  REGISTERED-PROVISIONAL status expires 12 month following its declaration in the “info” URI Registry, unless notification of public use is received from the requesting NAMESPACE AUTHORITY.

- Registration Duration

  An “info” URI Registration record remains in the “info” URI Registry as long as the NAMESPACE AUTHORITY is in good standing. If a NAMESPACE AUTHORITY is no longer in good standing, then the status of all registered NAMESPACE TOKENs in the “info” URI Registry

associated with said NAMESPACE AUTHORITY will, at the discretion of the "info" URI Registry Operator, be set to REGISTERED-INACTIVE. Additionally, the Registration record may be made inaccessible to the public.

VII. Characteristics of IDENTIFIERS managed within a given namespace.

IDENTIFIERS managed within a given namespace will have characteristics defined by the NAMESPACE AUTHORITY, constrained only by the character set constraints native to current network practice and characterized in [1].

VIII. Characteristics of an INFO URI Registration Record

An "info" URI Registration record will contain sufficient information to identify:

- Registered NAMESPACE TOKENs (string literals identifying the registered namespaces) [public]
- Contact information for the Namespace Authorities [public]
- Optional normalization methodology or parsing specifications for the INFO-IDENTIFIERS for a particular NAMESPACE TOKEN [public]
- Optional ancillary documentation or public contact information to support more effective utilization of the INFO-IDENTIFIERS [public]

All information not specified as public will be considered privileged business communication between the "info" URI Registry Operator and the NAMESPACE AUTHORITY and will not be sold, made available to third parties, or made publicly available except by written agreement. Collection and use of such information will be for the purpose of "info" URI Registry deployment and maintenance only.

IX. Rights and Responsibilities concerning NAMESPACE TOKEN and INFO-IDENTIFIERS

NAMESPACE AUTHORITIES may propose the token string to be associated with a proposed NAMESPACE TOKEN. In so doing, they assert that the namespace is in use or in formal planning stages by their organization, and that there are no known conflicts with existing service marks, trade marks, or other registered branding instruments.

The "info" URI Registry Operator reserves the right to reject a proposed NAMESPACE TOKEN and to propose alternates. Disagreements between the "info" URI Registry Operator and the NAMESPACE AUTHORITY will be mediated by the "info" URI Review Board.

NAMESPACE authorities agree to the public use of NAMESPACE TOKENs which they register, alone and in conjunction with INFO-IDENTIFIERS.

NAMESPACE AUTHORITIES do not, by the act of NAMESPACE TOKEN registration, forego any rights or interests in the use or commercial exploitation of intellectual property or systems based on the management of these NAMESPACE TOKENs beyond the requirement that these identifiers be publicly referenceable without fee.

The validation of INFO-IDENTIFIERs as actual instances of authentic identifiers is outside the scope and responsibility of this Registry.

X. Withdrawal of a Registered NAMESPACE TOKEN

A NAMESPACE AUTHORITY can initiate withdrawal of a registered NAMESPACE TOKEN at any time by written notification of discontinuance to the "info" URI Registry Operator. Such notification will trigger removal from public access of a designated NAMESPACE TOKEN record. Such withdrawal in no way restricts the public use of previously issued INFO-IDENTIFIERS.

The "info" URI Registry Operator can initiate withdrawal of a registered NAMESPACE TOKEN record for abrogation of Terms and Conditions specified in the Registration Agreement.

XI. Registration Fees

During the initial start-up and experimental period, no fees will be collected for NAMESPACE TOKEN registration. At some time in the future, NAMESPACE AUTHORITIES may be assessed cost-recovery fees for maintenance of the "info" URI Registry.

XII. Governance

Policy for the "info" URI Registry will be determined by a Governance Board to be constituted by NISO, the maintenance at a future time.

Policy for the "info" URI Registry will be available for public inspection and comment and will be established in accordance with principles of non-discriminatory openness and inclusiveness.

XIII. "info" URI Review Board

Review of "info" URI Registration requests will be conducted by an "info" URI Review Board.

XIV. References

[1] The "info" URI Scheme for Information Assets with Identifiers in Public Namespaces
<http://www.ietf.org/internet-drafts/draft-vandesompel-info-uri-01.txt>

Topic: Proposed terms for a collection description profile
Modified: 2004-02-25 15:44, Wednesday
Maintainer: Tom Baker
Latest version: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/terms-collection/
See also: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/

Shepherd: Andrew Wilson

SUMMARY (Tom)

Two proposals were submitted by Pete Johnston (who will attend the meeting in Bath):

- Dublin Core Collection Description Proposed Term : Provenance
  http://www.ukoln.ac.uk/metadata/dcmi/collection-provenance/

- Dublin Core Collection Description Proposed Term : Is Available At
  http://www.ukoln.ac.uk/metadata/dcmi/collection-isAvailableAt/

We will discuss these proposals and vote on approval.

These proposals raise some general issues, discussed in the email excerpts below: what it means to recommend the use of non-DCMI-maintained terms (instead of creating a DCMI term), and whether "collection-specific" terms properly belong in the main DCMI namespaces (or if not, where do they belong). Discussion of these issues overlaps with Topic 5 ("Scope of DCMI Namespaces"), see http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/dcmi-namespaces/.

Date: Mon, 9 Feb 2004 16:02:46 +0100
From: Thomas Baker <thomas.baker@bi.fhg.de>
To: DCMI Usage Board <dc-usage@jiscmail.ac.uk>

Pete's proposal will raise a basic issue we have not sufficiently addressed: What does it really mean for the UB to recommend the re-use of terms from other namespaces? We have done this before [1]. However, Pete argues that (for example) a MODS element -- an XML element which takes meaning from its context in an XML tree structure -- is fundamentally different from an RDF property usable in statements about resources [2].

Pete also wonders whether using a URI outside of its original context is consistent with good practice [3]. Should the owner of a URI be the one to make such a recommendation (e.g., LoC for MODS)?

[1] http://dublincore.org/usage/decisions/2002/2002-02.holdingLocation.shtml
[2] http://www.jiscmail.ac.uk/cgi-bin/wa.exe?A2=ind0401&L=dc-collections&T=0&F=&S=&P=4961

[3] http://www.w3.org/TR/webarch/#uri-ownership

Date: Mon, 9 Feb 2004 15:46:42 -0000
From: Pete Johnston <p.johnston@UKOLN.AC.UK>
Subject: Re: Bath -- term proposals versus "using" non-DCMI terms

---

> Pete also wonders whether using a URI outside of its original
> context is consistent with good practice [3]. Should the
> owner of a URI be the one to make such a recommendation
> (e.g., LoC for MODS)?

Just on this specific point of use outside of original context, I wasn't arguing against, say, DCMI recommending the use of a URI owned by LoC (or anyone else), as long as that use was in line with the assertions made by the URI owner about what that URI denoted.

So, if LoC says http://loc.gov/someproperty denotes an RDF property, and based on what LoC says about that property, DCMI thinks it seems useful, then it seems perfectly fine for DCMI to recommend the use of that property - as long as (a) DCMI's suggested use doesn't contradict LoC's recommended use for that property (e.g. if LoC says it should be used only to describe journal articles, then DCMI shouldn't say it can be used to describe museum objects, or cars, or ideas), and (b) DCMI is happy with LoC's policies for URI persistence and so on. (And probably some other conditions I haven't thought of!)

But I _was_ questioning whether DCMI (or indeed any other body) should, in the absence of explicit statements about what is denoted by a URI that it does not own, "make assumptions" about what that URI denotes.

Specifically for the case of the MODS location element, I couldn't see any clear evidence that a URI had been assigned to "the MODS location element" at all (an XML QName is not a URI) (though I'm aware of the decaration of intent made through the CORES resolution [1]).

And further the only information I can find about the MODS location element is a description of an XML element, not an RDF property, and those are fundamentally different things.

So I guess I'm - respectfully ;-) - questioning the line in the UB decision that says, 'the element "location" in MODS complies in a general way with Dublin Core principles', and is therefore appropriate for use in a DCAP.

[1] http://mirrored.ukoln.ac.uk/lis-journals/dlib/dlib/dlib/july03/baker/07baker.html

---

From: Pete Johnston <p.johnston@ukoln.ac.uk>
Date: Tue, 17 Feb 2004 12:55:55 -0000

I'm conscious that they both have (potentially at least) rather broader implications about the scope of DCMI activity, and particularly its relationship to the development of schemas and metadata vocabularies designed for the discovery of specific types of resource.

It seems to me that the DC CD WG may not be able to fulfil its current charter without some guidance on the second of these issues i.e. the assignment of URIs for terms that are conforming to the DC model and appropriate for use in an AP developed by a DCMI WG, but are not considered candidates for one of the DCMI Namespaces.

Looking again at section 3 of the UB process doc [1] it's not clear to me whether a property that was explicitly specific to the description of a Collection would be a candidate for a "conforming" term in the DCterms vocabulary (the description there talks about the scope of a domain or an implementer community rather than about the type of the resource).

As I say, if the answer is that collection-specific properties would not be "conforming" and therefore DCMI can not assign URIs for them, then the DC CD WG needs to look elsewhere for a URI owner. If on the other hand, a collection-specific property would be "conforming" and would be a candidate for the DCterms vocabulary, then we will make submissions to UB accordingly. (The proposals we have submitted this time are _not_ collection-specific, and we have hesitated to put forward terms that are explicitly collection-specific because it seemed unlikely that they would be accepted.)

There is perhaps a slight knock-on effect for the "ownership" issue, in that the DC CD AP as a whole might be a "DCMI product" but it may be using terms that were owned and administered by another party. That's perfectly possible of course, but something for us to be aware of.

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

Title: # Dublin Core Collection Description Proposed Term : Provenance

Creator:	Dublin Core Collection Description Working Group
Date Issued:	2004-02-10
Identifier:	http://www.ukoln.ac.uk/metadata/dcmi/collection-provenance/2004-02-10/
Replaces:	http://www.ukoln.ac.uk/metadata/dcmi/collection-provenance/2004-02-01/
Is Replaced By:	Not applicable
Latest Version:	http://www.ukoln.ac.uk/metadata/dcmi/collection-provenance/
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 element refinement, provenance.

Proposal

Name	**provenance**
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.

Examples	The provenance of a digital object: This file was created on 28 August 2002 by transformation from a TIFF image. The original TIFF image was purchased from the University of Sometown, Department of Special Collections and Archives in November 2001. The provenance of a physical object: The volume was presented to John Smith by Jones and Company on the occasion of his retirement from the company in 1888, and was located in Smith's private library until 1895, when it was sold to Wilson Booksellers of Liverpool. It was purchased for the National Library at auction in Manchester in June 1899. The provenance of a collection: The Smith collection was held by the Smith family in the library at Smith Hall. Letters written by John Smith and sent to Tom Jones remained in the custody of Jones until his death in 1966 when they were returned to Smith. The Smith collection was presented to the University Library as a gift by James Smith on 5 May 1973.
Type of term	Element refinement
Term qualified	Description, Dublin Core Metadata Element Set, v1.1 [dc:description] `http://purl.org/dc/elements/1.1/description`
Why needed	Information about changes in the ownership or custody of a collection may be important in helping a user to *select* a resource and/or, having selected a resource, to *interpret* the resource.
Working Group support	See the mailing list archives of the WG, especially Jan 2004 and Feb 2004.
Proposed status	Recommended

Related DCMI terms	None.
Related non-DCMI terms	The **International Standard for Archival Description (ISAD(G)), Second Edition (2000)** is a descriptive standard for archival records. It can be applied to units of description at any level from the collection (or fonds) to the individual item, and includes two related elements: • **3.2.3 Archival history** Purpose: To provide information on the history of the unit of description that is significant for its authenticity, integrity and interpretation. Rules: Record the successive transfers of ownership, responsibility and/or custody of the unit of description and indicate those actions, such as history of the arrangement, production of contemporary finding aids, re-use of the records for other purposes or software migrations, that have contributed to its present structure and arrangement. • **3.2.4 Immediate source of acquisition** Purpose: To identify the immediate source of acquisition or transfer. Rules: Record the source from which the unit of description was acquired and the date and/or method of acquisition if any or all of this information is not confidential. ISAD(G) does not itself provide a machine-readable binding. The proposed term would be used to capture information recorded in both of these ISAD(G) elements. The **Encoded Archival Description (EAD)** DTD is a standard for encoding archival finding aids using SGML or XML. It includes two XML elements: • **<custodhist> Provenance** Information about the chain of ownership of the materials being

	described, before they reached the immediate source of acquisition. Both physical possession and intellectual ownership can be described, providing details of changes of ownership and/or custody that may be significant in terms of authority, integrity, and interpretation. • **<acqinfo> Acquisition Information** The immediate source of the materials being described and the circumstances under which they were received. Includes donations, transfers, purchases, and deposits. These are XML elements, not RDF properties. The proposed term would be used to capture information recorded in both of these EAD elements. **DSpace** is a digital library system to capture, store, index, preserve, and redistribute the intellectual output of a university's research faculty in digital formats. The Dublin Core-based DSpace metadata schema includes an element refinement: • **dspace:provenance** Definition: The history of custody of the item since its creation, including any changes successive custodians made to it. The SIMILE project has developed RDF bindings for DSpace metadata, and an RDF property is described in http://web.mit.edu/simile/www/resources/history-harmony/history-design.pdf. However the DC CD WG expressed some concern about the persistence of URIs owned by DSpace.
Impact on applications	If existing DC applications for collection-level description capture this data, they probably do so as part of the value of a `dc:description` element. The introduction of a distinct term allows the metadata creator to separate this subset of information from the description of the scope and content of the collection. As the proposed term is defined as a refinement of `dc:description`, a statement using the proposed term can be "dumbed-down" to a statement using a `dc:description` element.

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

- Add example for digital resource.

Title: # Dublin Core Collection Description Proposed Term : Is Available At

Creator:	Dublin Core Collection Description Working Group
Date Issued:	2004-01-24
Identifier:	http://www.ukoln.ac.uk/metadata/dcmi/collection-isAvailableAt/2004-01-24/
Replaces:	http://www.ukoln.ac.uk/metadata/dcmi/collection-isAvailableAt/2004-01-18/
Is Replaced By:	Not applicable
Latest Version:	http://www.ukoln.ac.uk/metadata/dcmi/collection-isAvailableAt/
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 element refinement, isAvailableAt.

Proposal

Name	**isAvailableAt**
Label	Is Available At
Definition	The referenced resource provides access to the described resource.
Comment	The referenced resource may be a physical repository from which the described resource is supplied or a digital service that provides access to the described resource.

Examples	The Is Available At element refinement could be used to describe the relationship between: • a physical item and a repository that holds the item; • a collection of physical items and a repository that holds the collecton; • a collection of digital items and a Web site that provides access to the items; • a collection of metadata records and a Z39.50 target that provides access to those records
Type of term	Element refinement
Term qualified	Relation, Dublin Core Metadata Element Set, v1.1 [dc:relation] `http://purl.org/dc/elements/1.1/relation`
Why needed	The provision of access to an information resource is distinct from that information resource itself, and the two entities may be described as separate resources in distinct DC metadata descriptions. A single information resource may be available through multiple services. However, it is critical to be able to describe the relationship between these two resources. Having discovered an information resource, a user requires information on how to obtain access to it.
Working Group support	See the mailing list archives of the WG, especially Jan 2004 and Feb 2004. Also the report of the WG meeting at DC-2003.
Proposed status	Recommended
Related DCMI terms	Resource Identifier, Dublin Core Metadata Element Set, v1.1 [dc:identifier] `http://purl.org/dc/elements/1.1/identifier` Definition: 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. Example 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). A URL is an accepted and widely used value for dc:identifier. A URL is a type of URI that identifies a resource via a representation of its primary access mechanism (e.g., its network "location"), rather than by some other attributes it may have. http://www.w3.org/TR/uri-clarification/ However it remains the case that a URL is a URI and, used as a value of dc:identifier, it is still simply an identifier of a resource. It may identify a location or service because any type of resource (including a location or service) can be identified by a URI (including a URL) and described by a DC metadata description. However, it is not an inherent property of a URL that it identifies a location or service. By contrast, the value of the isAvailableAt property is a resource that provides access to the described resource.
Related non-DCMI terms	The **Metadata Object Description Schema (MODS)**, version 3.0, includes an XML element • **<location>** with "sub-elements": ○ **<physicalLocation>** ○ **<url>** The MODS User Guide has not yet been updated to describe these version 3.0 sub-elements. For version 2.0, the location element is described: "location" identifies the institution or repository holding the resource or from which it is available. It may be expressed as text or as a code with the authority attribute used to designate the source of the code. <location> may include specific

information to locate a resource within a collection, such as subcollection, or shelf number or it may just contain the name or code of an institution/repository.

<location> is equivalent to MARC 21 field 852 subfields $a, $b, $j, and $e. If information equivalent to these subfields is included, it is given in a string and is not parsed. Fuller holdings information may be included in the MODS <extension> element using an established XML holdings schema.

These are XML elements, not RDF properties. The proposed term would be used to capture information recorded in both of the sub-elements of MODS 3.0.

The **UK e-Government Metadata Standard (eGMS)** includes a metadata element:

- **Location**

  Definition: The physical location of the resource.

  Purpose: Enables the physical form of the resource to be found.

  Notes: Location will mainly be used for items held in a physical format, e.g. paper files.

  This is especially relevant for items listed in a metadatabase (a catalogue containing the metadata of resources but not the resources themselves). Metadatabases may refer to items not available in electronic format. It will also be valuable for electronic resources stored on physical media, e.g. magnetic tapes or CD-ROMs.

The eGMS Location element describes the relationship between a resource and its physical location. It is not used for the description of digital resources.

The **Australian Government Locator System (AGLS)** includes an element:

- **Availability**

  Definition: How the resource can be obtained or contact information for obtaining the resource.

  Comment: The Availability element is primarily used for non-electronic resources to provide information on how to obtain physical access to the resource.

The AGLS Availability element is used for the description of offline resources and (physical) services. In the description of a physical resource, it is used to describe a service that provides access to that physical resource; in the description of a service, it is used to describe the access conditions that apply to the service. It does not appear to be used for the description of digital resources.

The **New Zealand Government Locator System (NZGLS)** includes an element:

- **Availability**

  Definition: How the resource can be obtained or contact information.

  Purpose: The Availability element is mostly used for services and non-electronic resources, to provide information for searchers about physical access to the resource.

The NZGLS Availability element is used for the description of non-electronic resources and (physical) services. In the description of a physical resource, it is used to describe a service that provides access to that physical resource; in the description of a service, it is used to describe the access conditions that apply to the service. It does not appear to be used for the description of digital resources.

Impact on applications

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

- Expanded text under "Related DCMI terms".

Metadata associated with this resource: http://dublincore.org/documents/collection-isAvailableAt/index.shtml.rdf

Topic: Scope of DCMI Namespaces
Modified: 2004-02-25 15:44, Wednesday
Maintainer: Tom Baker
Latest version: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/dcmi-namespaces/
See also: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/

Shepherd: Andy Powell

-- http://www.ukoln.ac.uk/metadata/dcmi/dcmi-nmspc-scope/20040222/

> This document raises the question of what URIrefs should be assigned to terms proposed by DCMI WGs in the course of their work on DCAPs, when the use of those terms is limited to the description of a particular class of resource. Currently, although such terms may be conforming in terms of the Dublin Core model, it may be that they are considered inappropriate as candidates for inclusion in one of the DCMI Namespaces. Since this is the only means DCMI provides of assigning a persistent URIref, this means that the WG is unable to assign a DCMI-owned URI for the term, and must seek a third-party that can provide such a URIref.

Title:	**Scope of DCMI Namespaces**
Creator:	Pete Johnston
Creator:	Andy Powell
Date Issued:	2004-02-22
Identifier:	http://www.ukoln.ac.uk/metadata/dcmi/dcmi-nmspc-scope/20040222/
Replaces:	http://www.ukoln.ac.uk/metadata/dcmi/dcmi-nmspc-scope/20040220/
Is Replaced By:	Not applicable
Latest Version:	http://www.ukoln.ac.uk/metadata/dcmi/dcmi-nmspc-scope/
Status of Document:	This is a DCMI Working Draft.
Description of Document:	This document raises the question of what URIrefs should be assigned to terms proposed by DCMI WGs in the course of their work on DCAPs, when the use of those terms is limited to the description of a particular class of resource. Currently, although such terms may be conforming in terms of the Dublin Core model, it may be that they are considered inappropriate as candidates for inclusion in one of the DCMI Namespaces. Since this is the only means DCMI provides of assigning a persistent URIref, this means that the WG is unable to assign a DCMI-owned URI for the term, and must seek a third-party that can provide such a URIref.

DCMI Assignment of URIrefs and DCMI Namespaces

A central part of DCMI's role is to support the creation, development, maintainance and use of **DCMI terms**. All DCMI terms are grouped into vocabularies which DCMI refers to as **DCMI Namespaces**. Currently there are three DCMI Namespaces and each DCMI Namespace is identified by a URI reference.

As part of the process of creating a new DCMI term, that term is assigned a **name** and a **URI reference**

that provides a persistent, globally unique identifier for the term. Following the DCMI Namespace Policy, the URIref given to an individual term is determined by the vocabulary, the DCMI Namespace, to which the term is assigned. The URIref for the term is derived by appending the name of the term to the URIref of that DCMI Namespace. DCMI supports the assignment of these URIrefs with a policy statement guaranteeing their persistence as identifiers.[1].

Proposals for new terms typically emerge from the work of DCMI Working Groups who identify specific requirements for extending the DCMI vocabularies, and draft descriptions in the form required by the DCMI Usage Board. The Usage Board evaluates proposals according to a published set of criteria. [2].

DCMI Working Groups, DC Application Profiles and Resource-Type-Specific Properties

DCMI has emphasised its role in supporting cross-domain resource discovery. The properties (elements and element refinements) within the DCMI Namespaces are all applicable to a wide range of resource types; none are constrained to the description of a specific class of resource.

The Usage Board process document does *not* explicitly exclude the possibility of accepting proposals for properties that are specific to the description of one class of resource, but there is a widely-held perception that *only* properties that are applicable to a wide range of resource types are candidates for inclusion as elements or element refinements in DCMI Namespaces.

However, some Working Groups chartered by DCMI are working to develop DC-based metadata schemas for the description of specific types of resource, in the form of **DC Application Profiles**. The DC Collection Description WG is one such group [3, 4]. As part of this work, the group has suggested the use of properties that, while they are developed within the Dublin Core Grammatical Principles, are specific to the description of a Collection. For example, the term "Accrual Status" is used to provide

> A statement of accrual policy (closed, passive, active, partial/selective), accrual method (purchase, deposit) and accrual periodicity (closed, irregular, periodic). [5].

The semantics of this property dictate that it is meaningful only when applied to the description of a resource to which "accruals" can be made, i.e. a Collection.

In order for these properties to be usable by implementers, they must be assigned (persistent, globally unique) URIrefs, so that those URIrefs can be cited in statements made within "instance metadata" records. But the only procedure currently available within DCMI for the assignment of persistent identifiers is to submit the terms to Usage Board as candidates for a DCMI Namespace. If this option is closed, then the WG then faces the question of what URIrefs to assign to these properties.

Options Available

1. **No resource-type-specific terms permitted in AP**: If the only terms permitted in a DCAP are those from metadata vocabularies applicable to generic resource types, then the descriptive capacity of the AP is potentially limited. And in the case of the DC CD WG, the WG was chartered specifically to produce a DCAP for a single type of resource.
2. **No URIref assigned**: The WG might decide not to assign a URIref to the property at all and to provide only a human readable description of the property. However if the WG fails to assign a URIref to a property, the goal of semantic interoperability that the WG was chartered to achieve is lost, or at least compromised, as individual implementers are left to assign (possibly many different) URIrefs to refer to the property defined by the WG. The assignment of URIrefs to its properties is critical to the goals of the WG.
3. **Third-party-owned URI assigned**: If the WG itself can not assign persistent URIrefs, and DCMI UB can not assign the term to a DCMI Namespace, then the WG must seek a third-party authority who can provide this function, which in turn raises issues of trust in the capacity of that provider to deliver the persistence offered by DCMI.
4. **Inclusion of property in current "dcterms" DCMI Namespace**: It is widely understood that all properties in the dcterms vocabulary are applicable to a wide range of resource types, and introducing resource-type-specific properties would be potentially confusing to users. It may also open up the prospect of many proposals for other resource-type-specific properties, and would almost certainly require new policy for the Usage Board.
5. **Inclusion of property in a new DCMI Namespace**: If the use of the dcterms vocabulary is not appropriate, a separate vocabulary (DCMI Namespace) might be assigned for each resource-type-specific set of properties, with terms assigned URIrefs such as `http://purl.org/dc/collection/terms/accrualStatus`. Such a move would require the establishment of policies and procedures for the maintainance of those vocabularies by the Usage Board.

Issues Arising

If Option 3

What are implications for DCMI WGs? Is there a risk of WGs "stretching" the semantics of existing terms because it is diffcult to find an authority who can provide persistent URIrefs for new terms? Would different WGs find different authorities to provide URIrefs? How would users perceive the use of a range of URIs owned by different authorities?

If Option 4 or Option 5

All elements/element refinements issued by DCMI to date have been applicable to a broad range of resource types, and issuing resource-type-specific properties represents a shift in the scope of DCMI activity.

How would such a shift be perceived by users? Does it introduce a potential overlap with the activity of non-DCMI bodies creating metadata vocabularies? What would be the educational/documentational challenges?

What are the technical implications? Use of `rdfs:domain`?

Are there any limits to the resource-types for which DCMI might create resource-type-specific properties?

If Option 5

If new vocabularies/namespaces are created, how would these vocabularies be managed? By Usage Board in the same way as the current DCMI Namespaces?

References

[1] Powell, Andy and Harry Wagner. *Namespace Policy for the Dublin Core Metadata Initiative (DCMI)*, (November 2001)
http://dublincore.org/documents/dcmi-namespace/

[2] Hillman, Diane and Stuart A. Sutton. *DCMI Usage Board (UB) Administrative Processes*, (February 2003)
http://dublincore.org/usage/documents/process/

[3] *DC Collection Description Working Group*
http://dublincore.org/groups/collections/

[4] *Dublin Core Collection Description Application Profile*
http://www.ukoln.ac.uk/metadata/dcmi/collection-application-profile/

[5] *Term Proposal: [Collection] Accrual Status*
http://www.ukoln.ac.uk/metadata/dcmi/collection-accrualStatus/

Topic: DCMI Abstract Model
Modified: 2004-02-25 15:44, Wednesday
Maintainer: Tom Baker
Latest version: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/model/
See also: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/

Shepherd: Andy

SUMMARY (Tom)

Andy has revised the draft model:

http://www.ukoln.ac.uk/metadata/dcmi/abstract-model/

We need to discuss:

-- aspects of this model that relate to Usage Board
practice, such as the notion of Application Profile;

-- use of dcterms:URI as the encoding scheme for those parts of
encoded metadata that express a "value URI" as opposed to
a "value string" (see email below);

-- the relation of this document to other "foundational"
documents maintained by DCMI, especially
http://dublincore.org/usage/documents/principles/, which is
included in the main packet (see discussion below).

Notes from Tom on the relation between the DCMI Abstract Model (DAM)
and the existing DCMI Grammatical Principles (DGP) at
http://dublincore.org/usage/documents/principles/.

I see two slightly different needs:

1) The need to present this new model to the general public
in a self-contained and readable form.

2) The need for DCMI to have a set of URI-identified documents
that makes optimal sense from the standpoints of citation
and maintenance.

Need #1 is (almost) met by the existing draft, which
contains almost all of the elements one would need for a

self-contained presentation to a broad public. For this, in my opinion, it still needs a narrative introduction -- one which wraps the model in some historical context and briefly states how the present document relates to earlier documents. With such a context, the article would basically be ready for D-Lib Magazine (let's call this "DLIB-DAM" as opposed to a "DCMI-DAM" to be maintained as a DCMI document).

In terms of Need #2, however, and with respect to other DCMI-maintained documents, the DAM "package" (model plus Appendixes) does not seem ideal:

-- The existing two-page "DCMI Grammatical Principles" (http://dublincore.org/usage/documents/principles) is almost completely subsumed in Section 8 (Terminology) of DAM.

   The only things missing from DAM with respect to DGP are:
   -- a two-sentence scope statement that explicitly gets the definition of "term" from the DCMI Namespace Policy;
   -- a one-sentence definition of "appropriate value";
   -- a three-sentence discussion of Vocabulary Terms (with the recent addition of "Broader Than" and "Narrower Than").
   -- a one-sentence definition of "Application Profile" (along the lines of the one I suggested for DAM);
   -- a short footnote acknowledging a historical shift towards seeing Element Refinements as Properties.

   Having to maintain DAM and DGP in sync creates potential error and confusion (and extra work). The solution could be either to delete DAM's Section 8 ("Terminology") and replace it with a pointer to DGP -- or vice versa. I am inclined to the latter, i.e. to delete the DGP document (redirecting its URI) and add some (but not all!) of the missing bits to DAM. Then the terminology section (Section 8) could perhaps be moved up to the front of DCMI-DAM -- boring for a D-Lib article but good practice for a specification.

-- Rather than leave the comparisons of RDF, XML, and XHTML encoding specs against the model in the DCMI-DAM document itself, I might sense either:

   1) to fold each of the comparisons into the Specification in question;
   2) to break out each comparison as a separate document with a citable URI and (in the longer term) subject to a

separate processe of revision and approval; or
3) to bundle all of the comparisons into a single, citable document (but not as an Appendix to the model itself).

On the other hand, the appendix on RDF as it is currently written can help people who already know RDF understand the Abstract Model itself, so perhaps it should remain in the Appendix even if yet-to-be-written comparisons for DCMI's two RDF-related Specifications reside elsewhere.

I do not have a strong opinion about these options.

-- The note about structured values is on one level an abstract typology, but it is also framed as a response to the state of discussion as of the time of writing. One could perhaps split it out as a separate document with a separately citable URI, though it is only one page long so I do not feel strongly about this.

Note that following the suggestions would have the effect of making DCMI-DAM document shorter (just five pages) and turning the existing two-page "DCMI Grammatical Principles" into a redirect to DCMI-DAM.

Date: Wed, 25 Feb 2004 10:49:27 +0000
From: Andy Powell <a.powell@UKOLN.AC.UK>

When a vocabulary term gets a URI assigned to it, it changes from being a 'value string' to being a 'value URI' and the encoding needs to change to reflect that. So for example in XML, instead of

```
<dc:subject xsi:type="dcterms:DDC">Internet</dc:subject>
```

or

```
<dc:subject xsi:type="dcterms:DDC">004.678</dc:subject>
```

depending on your preference for numbers or words, the encoding would change to something more like

```
<dc:subject xsi:type="dcterms:URI">info:ddc/22/eng//004.678</dc:subject>
```

As usual, the RDF/XML encoding handles this rather better.

I'll write up something longer about this before the meeting. This is just to flag up the issue.

Title: # DCMI Abstract Model

Creator: Andy Powell
UKOLN, University of Bath, UK
Mikael Nilsson
KMR Group, CID, NADA, KTH (Royal Institute of Technology), Sweden
Ambjörn Naeve
KMR Group, CID, NADA, KTH (Royal Institute of Technology), Sweden
Pete Johnston
UKOLN, University of Bath, UK

Date Issued: 2004-02-04
Identifier: http://dublincore.org/documents/2004/02/04/abstract-model/
Replaces: http://www.ukoln.ac.uk/metadata/dcmi/abstract-model/2004-01-16/
Is Replaced By: Not applicable
Latest Version: http://dublincore.org/documents/abstract-model/
Status of Document: This is a DCMI Working Draft.
Description of Document: This document describes an abstract model for DCMI metadata descriptions.

Table of contents

1. Introduction

This document specifies an abstract model for DCMI metadata descriptions [DCMI]. The primary purpose of this document is to provide a reference model against which particular DC encoding guidelines can be compared. To function well, a reference model needs to be independent of any particular encoding syntax. Such a reference model allows us to gain a better understanding of the kinds of descriptions that we are trying to encode and facilitates the development of better mappings and translations between different syntaxes.

2. DCMI abstract models

The abstract model of the *resources* being described by DCMI metadata descriptions is as follows:

- Each *resource* has zero or more *properties*.
- Each *property* has one or more *values*.
- Each *value* is a *resource* (the physical or conceptual entity that is associated with a *property* when it is used to describe a *resource*).
- Each *resource* may be a member of one or more *classes*.
  (Note that where the *resource*is the *value*of a *property*, the *class*is referred to as a *vocabulary encoding scheme*.)
- Each *property* and *class* have some declared *semantics*.
- Each *class* may be related to one or more other *classes* by a refines (sub-class) relationship (where the two *classes* share some *semantics* such that all *resources* that are members of the *sub-class* are also members of the related *class*.
- Each *property* may be related to exactly one other *property* by a refines (sub-property) relationship (where the two *properties* share some *semantics* such that all valid *values* of the *sub-property* are also valid *values* of the related *property*).

The abstract model of DCMI metadata descriptions is as follows:

- A *description* is made up of one or more *statements* (about one, and only one, *resource*) and zero or one *resource URI* (a URI that identifies the *resource* being described).
- Each *statement* is made up of a *property URI* (a URI that identifies a *property*), zero or one *value URI* (a URI that identifies a *value* of the *property*), zero or one *encoding scheme URI* (a URI that identifies the *class* of the *value*) and zero or more *value representations* of the *value*.
- Each *property* is an attribute of the *resource* being described.
- Each *property URI* may be repeated in multiple *statements*.
- The *value representation* may take the form of a *value string*, a *rich value* or a *related description*.
- Each *value string* is a simple, human-readable string that represents the *value* of the *property*.
- Each *value string* may have an associated *encoding scheme URI* that identifies a *syntax encoding scheme*.
- Each *value string* may have an associated *value string language* that is an ISO language tag (e.g. en-GB).
- Each *rich value* is some marked-up text, an image, a video, some audio, etc. or some combination thereof that represents the *resource* that is the *value* of the *property*.
- Each *related description* is a description of (i.e. some metadata about) the *resource* that is the *value* of the *property*.

The italicised terms used above are defined in the terminology section below. A number of things about the model are worth noting:

- A *related description* describes a related *resource* and is therefore not part of the *description* - for example, a *related description* may provide metadata about the person that is the creator of the described *resource*.
- *Syntax encoding schemes* are also known as 'datatypes' in some contexts.
- In DCMI metadata *descriptions*, the *class* of the *resource* being described is normally indicated by the *value* of the DC Type *property*.

The DCMI abstract models for resources and descriptions are represented as UML class diagrams [UML] in figures 1 and 2.

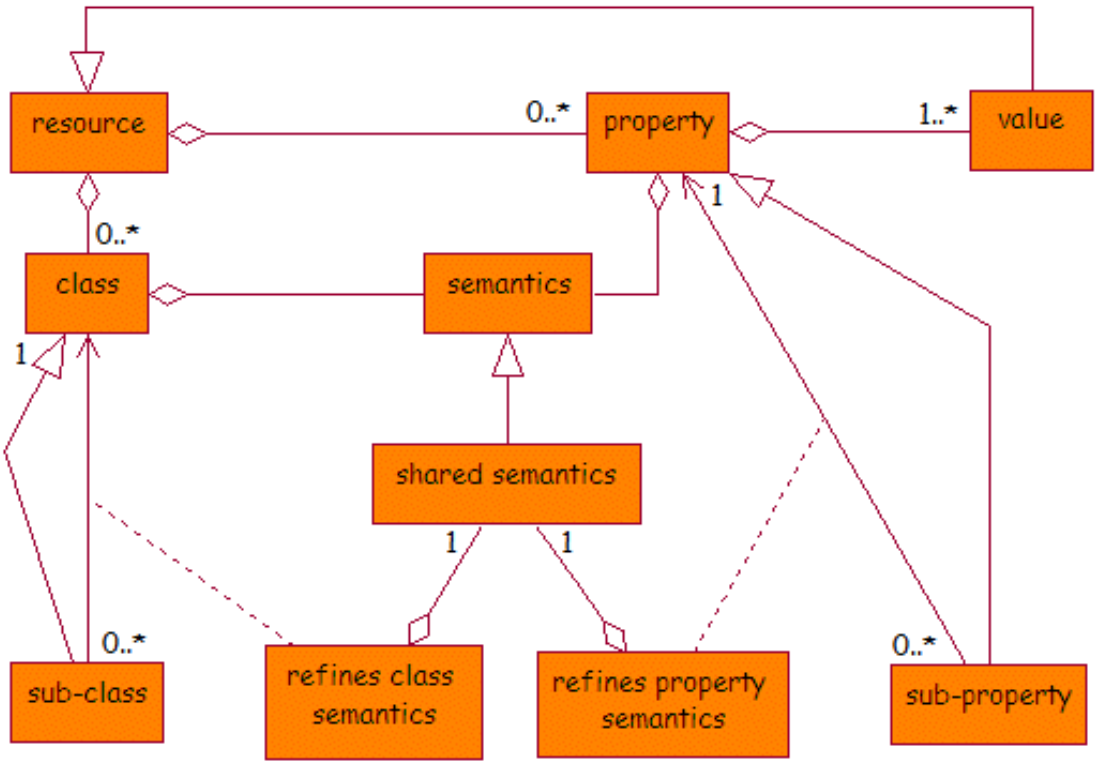
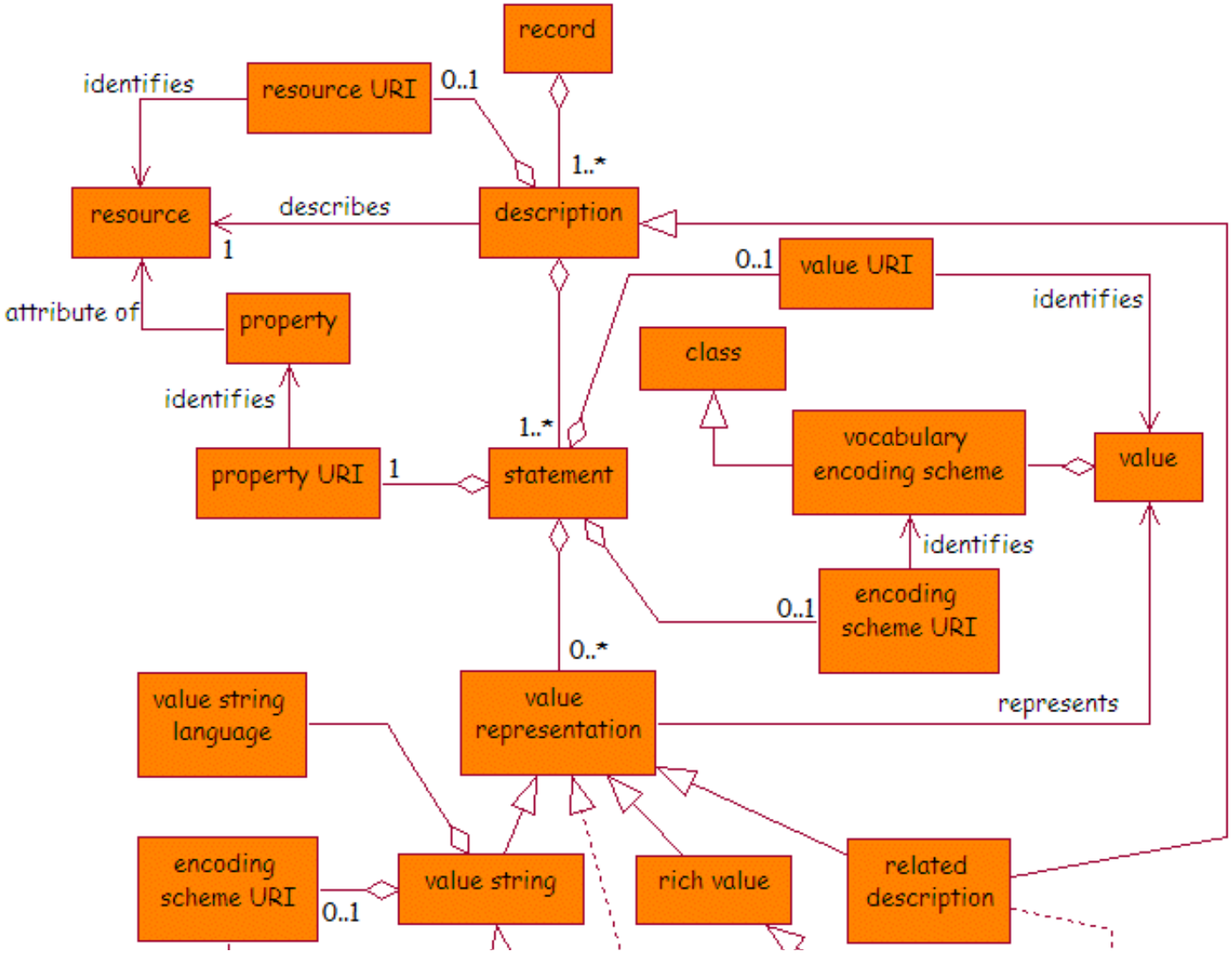


Figure 1 - the DCMI resource model



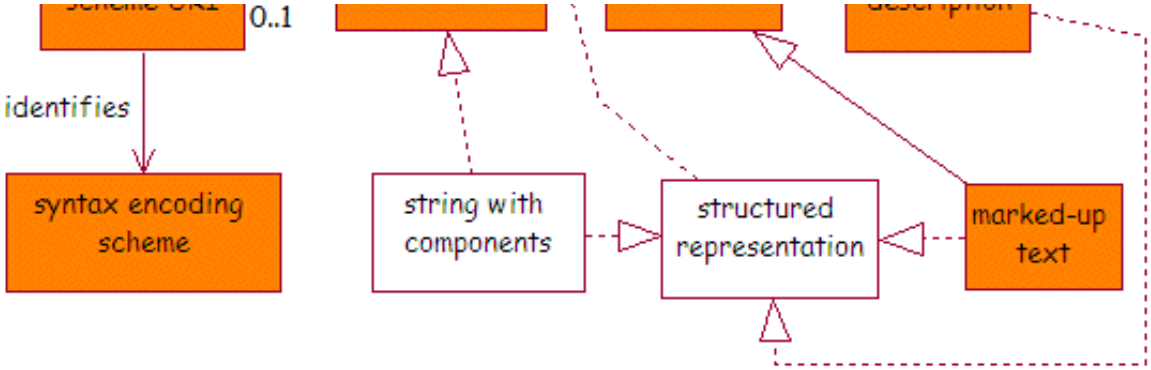


Figure 2 - the DCMI description model

Readers that are not familiar with UML class diagrams should note that lines ending in a block-arrow should be read as 'is' or 'is a' (for example, 'a *vocabulary encoding scheme* is a *class*') and that lines starting with a block-diamond should be read as 'contains a' or 'has a' (for example, 'a *statement* contains a *property URI*'). Other relationships are labelled appropriately. The classes represented by the clear boxes are not mentioned explicitly in the textual description of the abstract model above but are discussed in Appendix A. Note that the UML modelling used here shows the abstract model but is not intended to form a suitable basis for the development of DCMI software applications.

3. Records and descriptions

The abstract model described above indicates that each *property* used in a *description* must be an attribute of the *resource* being described. This is commonly referred to as the 1:1 principle - the principle that a DCMI metadata *description* describes one, and only one, resource.

However, real-world metadata applications tend to be based on loosely grouped sets of *descriptions* (where the described *resources* are typically related in some way). Such sets are often referred to as metadata *records*. For example, a metadata *record* might comprise *descriptions* of both a painting and the artist. Furthermore, it is often the case that the record will also contain a description about the metadata *record* itself (sometimes refered to as 'admin metadata' or 'meta-metadata').

This document defines a DCMI metadata *record* as follows:

- A DCMI metadata *record* is a set of one or more *descriptions* about one or more related *resources* instantiated according to one of the DCMI encoding guidelines (XHTML meta tags, XML, RDF/XML, etc.) [DCMI-ENCODINGS].

4. Values

A DCMI metadata *value* is the physical or conceptual entity that is associated with a *property* when it is used to describe a *resource*. For example, the *value* of the DC Creator *property* is a person, organisation or service - a physical entitiy. The *value* of the DC Date *property* is a point in time - a conceptual entity. The *value* of the DC Coverage *property* may be a geographic region or country - a physical entity. The *value* of the DC Subject *property* may be a concept - a conceptual entity - or a physical object or person - a physical entity. Each of these entities is a *resource*.

The *value* may be identified using a *value URI*; the *value* may be represented by one or more *value strings* and/or *rich values*; the *value* may have some *related descriptions* - but the *value* is a *resource*.

5. Simple and qualified DC

A Dublin Core Application Profile [DCAP] is a declaration specifying, at a minimum, which *properties* are used within a particular metadata application. Optionally, an application profile may describe how those *properties* have been constrained or adapted for

particular purposes.

The commonly used notions of simple and qualified Dublin Core are application profiles for two types of *records* based on DCMI terms. These notions can be described using the abstract model as follows:

- A simple DC record is a *record* that:
  - conforms to the abstract model,
  - comprises only a single *description*,
  - uses only the 15 *properties* in the Dublin Core Metadata Element Set [DCMES],
  - makes no use of *value URIs*, *encoding schemes*, *rich values* or *related descriptions*.
- A qualified DC record is a *record* that:
  - conforms to the DCMI abstract model,
  - contains at least one *property* taken from the DCMI Metadata Terms recommendation [DCTERMS]).

6. Dumb-down

The process of translating a qualified DC metadata record into a simple DC metadata record is normally referred to as 'dumbing-down'. The process of dumbing-down can be separated into two parts: property dumb-down and value dumb-down. Furthermore, each of these processes can be be approached in one of two ways. Informed dumb-down takes place where the software performing the dumb-down algorithm has knowledge built into it about the *property* relationships and *values* being used within a specific DCMI metadata application. Uninformed dumb-down takes place where the software performing the dumb-down algorithm has no prior knowledge about the *properties* and *values* being used.

Based on this analysis, it is possible to outline a 'dumb-down algorithm' matrix, shown below:

	Element dumb-down	**Value dumb-down**
Uninformed	Ignore any *property* that isn't in the Dublin Core Metadata Element Set [DCMES].	Use *value URI* (if present) or *value string* as new *value string*.
Informed	Recursively resolve sub-property relationships until one of the 15 properties in the Dublin Core Metadata Element Set [DCMES] is reached, otherwise ignore.	Use knowledge of the *related descriptions* or the *value string* to create a new *value string*.

In all cases, the dumb-down algorithm should also:

- ignore any *related descriptions* and *rich values*,
- ignore any *encoding scheme URIs*.

Note that software should make use of the DCMI term declarations represented in RDF schema language [DC-RDFS] and the DC XML namespaces [DC-NAMESPACES] to automate the resolution of sub-property relationships.

7. Encoding guidelines

Particular encoding guidelines (HTML meta tags, XML, RDF/XML, etc.) [DCMI-ENCODINGS] do not need to encode all aspects of the abstract model described above. However, DCMI recommendations that provide encoding guidelines should refer to the DCMI abstract model and indicate which parts of the model are encoded and which are not. In particular, encoding guidelines

should indicate whether any *rich values* or *related descriptions* associated with a *statement* are embedded within the *record* or are encoded in a separate *record* and linked to it using a URI.

Appendices B, C and D below provide a summary comparison between the abstract model and the RDF/XML, XML and XHTML encoding guidelines.

8. Terminology

This document uses the following terms:

resource
: A *resource* is anything that has identity. Familiar examples include an electronic document, an image, a service (e.g., "today's weather report for Los Angeles"), and a collection of other *resources*. Not all *resources* are network "retrievable"; e.g., human beings, corporations, concepts and bound books in a library can also be considered *resources*.

resource URI
: A *resource URI* is a URI that identifies a single *resource*.

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

property URI
: A *property URI* is a URI that identifies a single *property*.

element
: Within DCMI, *element* is typically used as a synonym for *property*. However, it should be noted that the word element is also commonly used to refer to a structural markup component within an XML document.

element refinement
: An *element refinement* is a *property* of a *resource* that shares the meaning of a particular DCMI *property* but with narrower semantics. Since *element refinements* are *properties*, they can be used in metadata *descriptions* independently of the *properties* they refine. In DCMI practice, an *element refinement* refines just one parent DCMI *property*.

description
: A *description* is made up of one or more *statements* about one, and only one, *resource*.

statement
: A *statement* is made up of a *property URI* (a URI that identifies a *property*), zero or one *value URI* (a URI that identifies a *value* of the *property*), zero or one *encoding scheme URI* (a URI that identifies the *class* of the *value*) and zero or more *value representations* of the *value*.

record
: A *record* is a set of one or more *descriptions* about one or more related *resources*, instantiated according to one of the DCMI encoding guidelines (XHTML meta tags, XML, RDF/XML, etc.) [DCMI-ENCODINGS].

value
: A *value* is the physical or conceptual entity that is associated with a *property* when it is used to describe a *resource*.

value URI
: A *value URI* is a URI that identifies the *value* of a *property*.

value representation
: A *value representation* is a surrogate for (i.e. a representation of) the *value*.

value string
: A *value string* is a simple string that represents the *value* of a *property*. In general, a *value string* should not contain any *marked-up text*.

value string language
The *value string language* indicates the language of the *value string*.

vocabulary encoding scheme
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 the Library of Congress Subject Headings.

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.

encoding scheme
Encoding scheme is the generic term for *vocabulary encoding scheme* and *syntax encoding scheme*.

encoding scheme URI
An *encoding scheme URI* is a URI that identifies an *encoding scheme*. For all DCMI recommended *encoding schemes*, the URI is constructed by concatenating the name of the *encoding scheme* with the DCTerms namespace URI.

marked-up text
A string that contains HTML, XML or other markup (for example TeX) and that is associated with the *value* of a *property*.

rich value
Some *marked-up text*, an image, a video, some audio, etc. (or some combination thereof) that is associated with the *value* of a *property*.

related description
A *related description* is a *description* of a *resource* that is related to the *resource* being described.

qualifier
Qualifier was the generic term used for the terms that are now usually referred to specifically as *element refinements* or *encoding schemes*.

structured value
Structured value is the generic term for the following:

- A *value string* that contains machine-parsable component parts (and which has an associated *syntax encoding scheme* that indicates how the component parts are encoded within the string).
- Some *marked-up text*.
- A *related description*

References

DCMI
Dublin Core Metadata Initiative
<http://dublincore.org/>

UML
The Unified Modeling Language User Guide
Grady Booch, James Rumbaugh and Ivar Jacobson, Addison-Wesley, 1998

DCTERMS
DCMI Metadata Terms
<http://dublincore.org/documents/dcmi-terms/>

DCMES
Dublin Core Metadata Element Set, Version 1.1: Reference Description
<http://dublincore.org/documents/dces/>

DCMI-ENCODINGS

DCMI Encoding Guidelines
<http://dublincore.org/resources/expressions/>

DCAP
DCMI Usage Board Review of Application Profiles
<http://dublincore.org/usage/documents/profiles/>

DC-RDFS
DCMI term declarations represented in RDF schema language
<http://dublincore.org/schemas/rdfs/>

DC-NAMESPACES
Namespace Policy for the Dublin Core Metadata Initiative (DCMI)
<http://dublincore.org/documents/dcmi-namespace/>

Acknowledgements

Thanks to Tom Baker, the members of the DC Usage Board and the members of the DC Architecture Working Group for their comments on previous versions of this document.

Appendix A - A note about structured values

This appendix discusses 'structured values', as they are used in DC metadata applications at the time of writing.

Many existing applications of DC metadata have attempted to encode relatively complex descriptions (i.e. descriptions that contain more than simply a property and its string value). These attempts have been loosely referred to as 'structured values'. It is possible to identify a number of different kinds of structured values. Four are enumerated below. The first two of these are recommended by the DCMI, in the sense that there are a number of existing encoding schemes that define values that conform to these definitions of structured values. The latter two are not currently recommended, but it is likely that they are in fairly common usage across metadata applications worldwide.

Labelled strings

These are strings that contain explicitly labelled components. Examples of this kind of structured value include:

DCSV
and the various DCMI syntax encoding schemes built on it - Period, Point and Box. An example of the use of DCSV in Period is:

```
<meta name="dcterms:temporal"
      scheme="dcterms:Period"
      content="start=Cambrian period; scheme=Geological timescale; name=Phanerozoic
Eon;" />
```

vCard
for example:

```
<meta name="dc:creator"
```

```
        content="BEGIN:VCARD\nORG:University of Oxford\nEND:VCARD\n" />
```

Note that vCard is not currently a DCMI recommended encoding scheme.

Unlabelled strings

These are strings that contain implicit components within the string, i.e. the components are determined based solely on their position within the string. Examples of this kind of structured value include:

W3CDTF
: the date-time format used within most DC metadata. For example:

```
<meta name="dc:date"
      scheme="dcterms:W3CDTF"
      content="2003-06-10" />
```

Marked-up text

These are strings containing 'presentational' or other markup, for example adding paragraph breaks, superscripts or chemical/mathematical markup to a dc:description. It is possible to characterise various kinds of markup as follows:

- Markup based on a version of HTML.
- Markup based on other XML-based languages such as CML and MathML.
- Non-XML markup languages such as TeX.

Related resource descriptions

These are metadata descriptions that describe a second resource (i.e. not the resource being described by the DC description). For example, a related description associated with the value of dc:creator could contain a complete description of the resource author (including birthday, eye-colour and favourite beverage if desired!).

In the past, 'related resource descriptions' have tended to be encoded using XML, vCard (see above) or by inventing multiple 'refinements' of DCMES properties (for example DC.Creator.Address). The RDF/XML encoding of DC (see below) provides us with a more thorough modelling of related metadata records through the use of multiple linked nodes in an RDF graph.

In DC metadata records, the following properties (and their element refinements) are used to provide the name or identifier of a second resource that is related to the resource being described:

- dc:creator
- dc:contributor
- dc:publisher
- dc:relation
- dc:source

In the case of the first three, this is typically done by providing the name (or in some cases the name and a small amount of additional information in order to better identify the person or organisation) of the related resource as the value string.

In the case of the last two, this is typically done by providing the URI (or some other identifier) of the related resource as the value URI. However, where no identifier is available, the name of the related resource can be provided instead (or as well) using the value string.

It should be noted that the value strings of these properties (and their element refinements) are not intended to be used to provide full descriptions of the related resource.

Summary

The categories outlined above are not watertight and there are certainly overlaps between them. For example, labelled strings can be viewed as a type of non-XML markup language. In addition, there will be cases where marked-up text (e.g. MathML) can be viewed as a related resource description.

Nevertheless, the purpose of the categorisation used here is to try and analyse existing usage of complex metadata structures within current DC metadata applications. In the context of the abstract model proposed here, all the types of structured values outlined above form part of the DCMI abstract model:

- A labelled string should be treated as a *related description* (though it should be noted that DCSV and the various DCMI syntax encoding schemes built on it - Period, Point and Box - are currently encoded as *value strings* with an appropriate *syntax encoding scheme*).
- An unlabelled string should be treated as a *value string* with an appropriate *syntax encoding scheme*.
- Marked-up text should be treated as a *rich value*.
- A related resource description should be treated as a *related description*.

Appendix B - The abstract model and RDF

This appendix discusses the relationship between the DCMI abstract model and the Resource Description Framework (RDF).

RDF currently provides DCMI with the richest encoding environment of the available encoding syntaxes. It is therefore worth taking a brief look at how the abstract model described here compares with the RDF model.

Note that the intention here is not to provide a full and detailed description of how to encode DC metadata records in RDF. Instead, three simple examples of the use of DC in RDF are considered.

Example 1: dc:creator

Figure 3 shows a simple RDF graph (and the RDF/XML document that represents it). The graph shows a resource with a single property (dc:creator). The *value* of the property is a second (blank) node, representing the creator of the resource. This second blank node has several properties, used to describe the creator, and an rdfs:label property that is used to provide the *value string* for the dc:creator property.

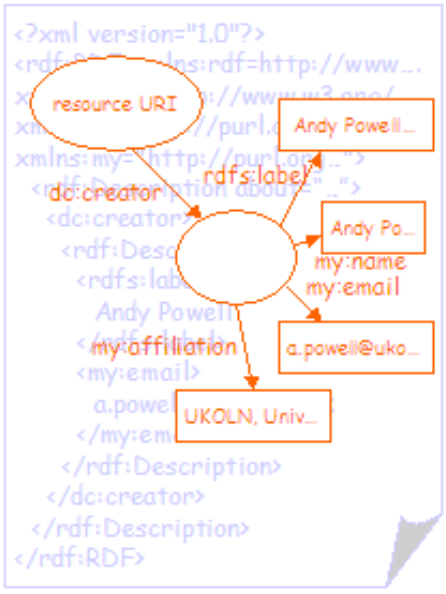


Figure 3

Figure 4 shows the same information separated into two graphs. In this case the *related description* that describes the creator has been more clearly separated from the description of the resource by moving it into a separate RDF/XML document. In order to do this, the node representing the *value* has been assigned a *value URI*, allowing the two nodes in the two RDF/XML documents to be treated as representing the same thing.

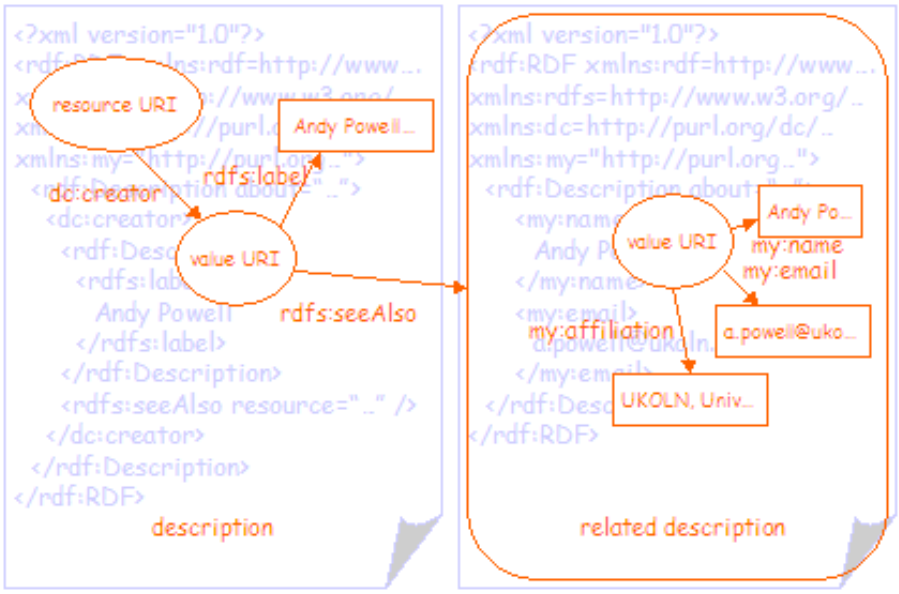


Figure 4

The *related description* in the second RDF/XML document is linked to the first using the rdfs:seeAlso property and the URI of the RDF/XML document. Note that it is not strictly necessary to separate the two graphs in this way; it is perfectly valid to represent the second graph as a sub-graph of the first, as shown in figure 3. However, for the purposes of this document, the two graphs have been separated in order to more clearly differentiate the *description* from the *related description*. In some cases it will be good practice to facilitate this separation anyway. For example, in order to serve the second graph from a directory service of some kind.

Example 2: dc:subject

Figure 5 shows a second simple RDF graph (and the RDF/XML document that represents it). The graph shows a resource with a single property (dc:subject). The *value* of the property is a second (blank) node, representing the subject of the resource. This second blank node has an rdfs:label property that is used to provide the *value string* for the dc:subject property, an rdf:value property that is used to provide the classification scheme notation and an rdf:type property to provide the encoding scheme URI.

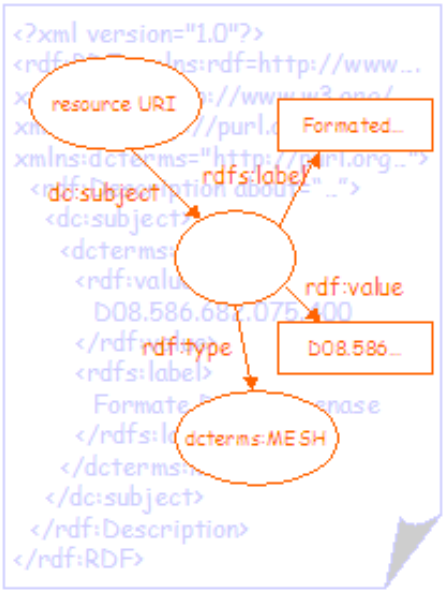


Figure 5

Figure 6 shows the same information separated into two graphs. In this case the *related description* that describes the subject has been more clearly separated from the description of the resource by moving it into a separate RDF/XML document. In order to do this, the node representing the *value* has been assigned a *value URI*, allowing the two nodes in the two RDF/XML documents to be treated as representing the same thing.

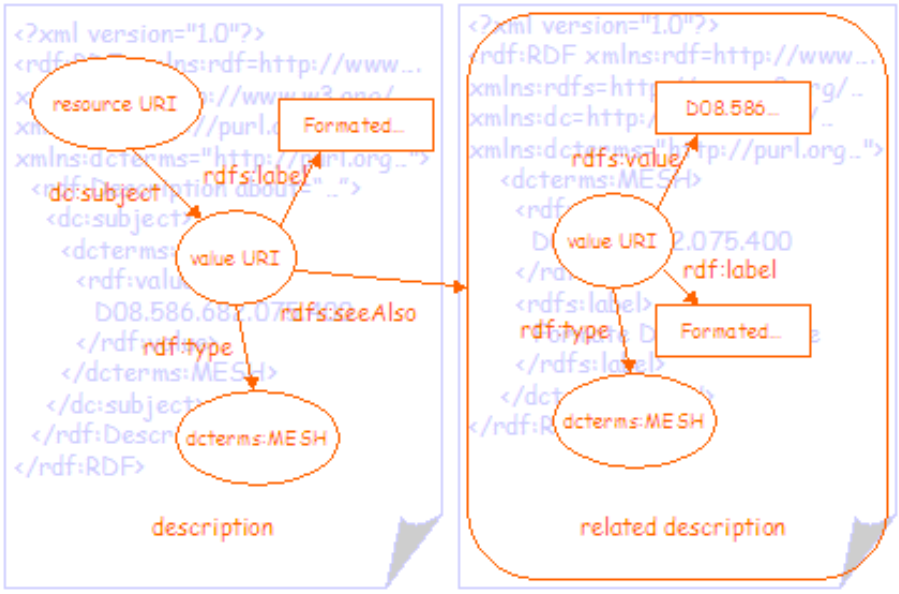


Figure 6

The *related description* in the second RDF/XML document is linked to the first using the rdfs:seeAlso property and the URI of the RDF/XML document. Note that it is not strictly necessary to separate the two graphs in this way; it is perfectly valid to represent the second graph as a sub-graph of the first, as shown in figure 5. However, for the purposes of this document, the two graphs have been separated in order to more clearly differentiate the *description* from the *related description*. In some cases it will be good practice to facilitate this separation anyway. For example, in order to serve the second graph from a terminology service of some kind.

Example 3: dc:description

Figure 7 shows a third simple RDF graph (and the RDF/XML document that represents it). The graph shows a resource with a single property (dc:description). The *value* of the property is a second (blank) node with an rdfs:label property that is used to provide the *value string* for the dc:description property. The second node also has an rdfs:seeAlso property that links to a *rich value* - in this case some HTML marked-up text that provides a richer representation of the description.

Note that it is possible to embed the marked-up text within a single RDF graph (using rdf:parseType="Literal"). However, this is not shown here.

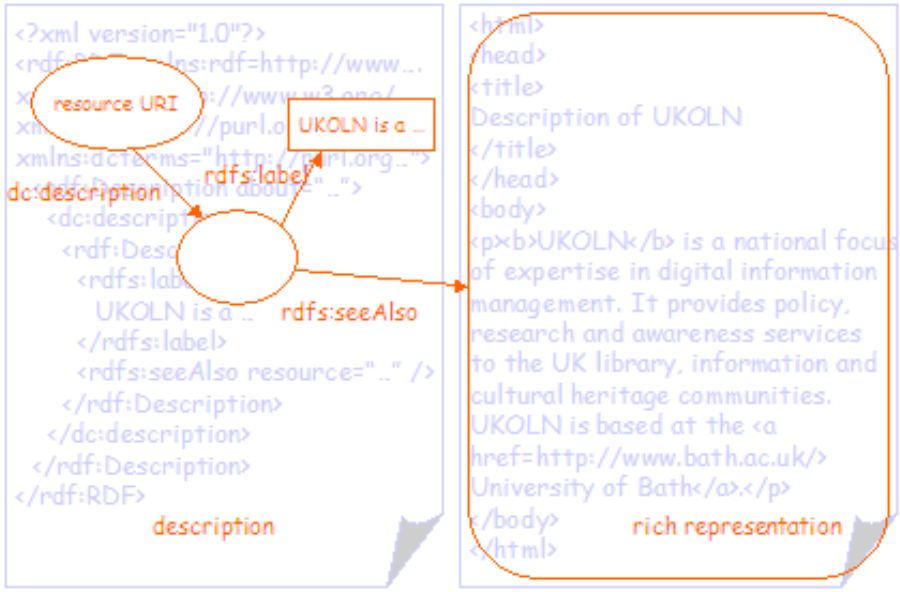


Figure 7

Summary

By re-visiting the second figure from example 2 (figure 6) it is possible to layer the terminology used in the abstract models above over the RDF graph.

All aspects of the DCMI abstract model are supported by the RDF encoding guidelines.

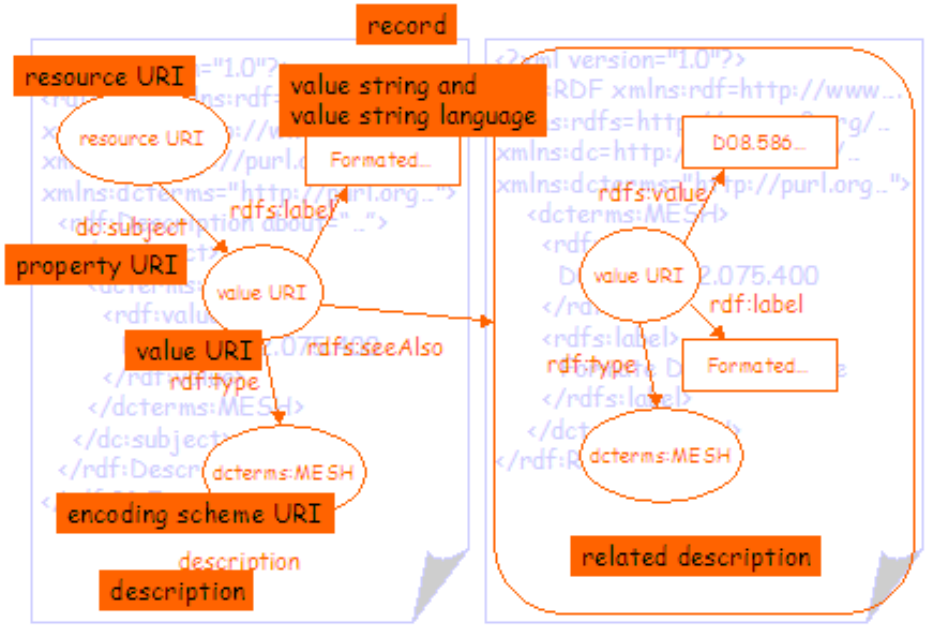


Figure 8

Appendix C - The abstract model and XML

This appendix compares the DCMI abstract model with the Guidelines for implementing Dublin Core in XML DCMI recommendation.

Simple DC

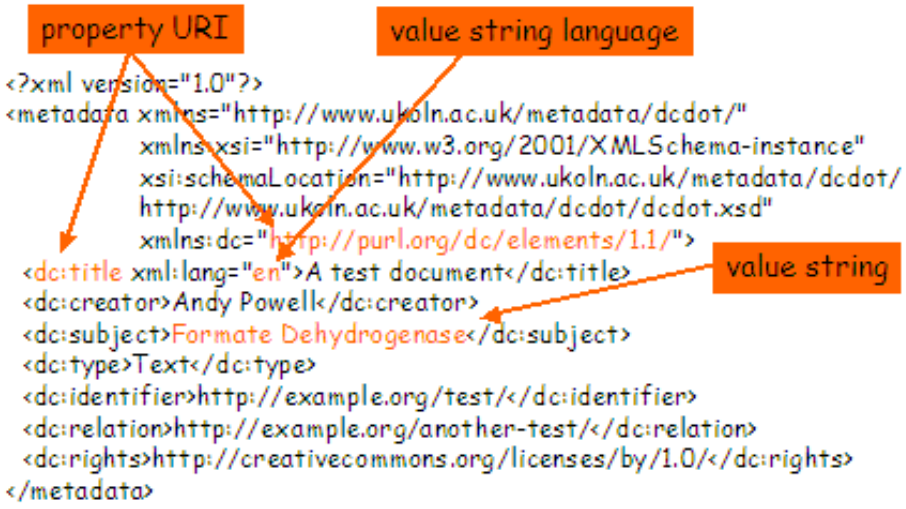


Figure 9

Figure 9 shows an example simple DC description encoded according to the XML guidelines above. The example shows how the encoding supports the *property URI*, *value string* and *value string language* aspects of the DCMI abstract model. It should be noted that all the *values* that are encoded in this syntax are represented by *value strings*, even those that look, to the human reader, as though they are URIs.

Qualified DC

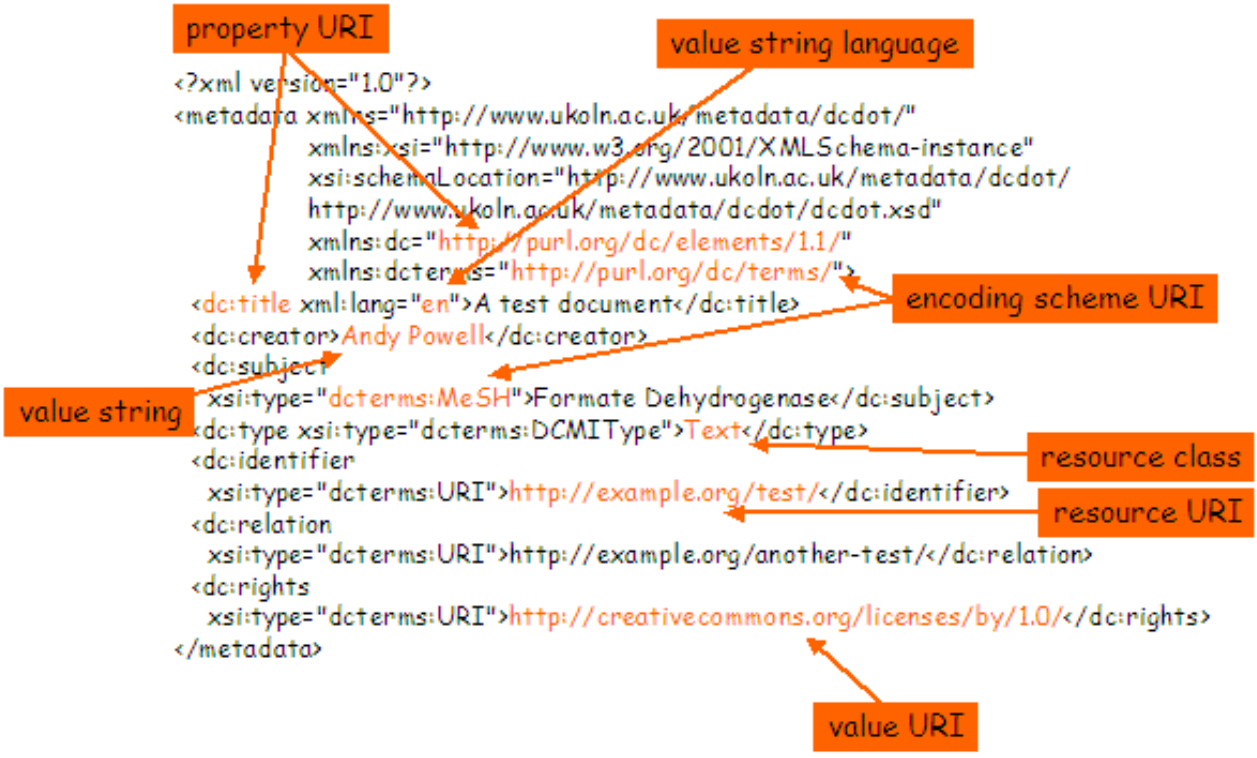


Figure 10

Figure 10 shows an example qualified DC description encoded according to the XML guidelines above. This example shows how the encoding supports the *property URI*, *value string*, *value string language*, *value URI*, *encoding scheme URI*, *resource URI* and *resource class* aspects of the DCMI abstract model. Note that the 'dcterms:URI' *encoding scheme* is used to indicate that the content of the XML element is a *value URI*. Note also that, although the *resource class* is indicated, the *class URI* is not encoded anywhere in this description.

Summary

The following aspects of the DCMI abstract model are supported by the Guidelines for implementing Dublin Core in XML recommendation:

- *properties*
- *property URIs*
- *value strings*
- *value string languages*
- *value URIs*
- *encoding schemes*
- *encoding scheme URIs*
- *resource URIs*
- *resource classes*

The following aspects of the DCMI abstract model are not supported:

- *rich values*
- *related descriptions*
- *property*/sub-property relationships
- *resource class URIs*

The following constraints apply:

- Each *property* may have one *value string* (but not more than one) or a *value URI* but not both.
- *Vocabulary encoding schemes* and *syntax encoding schemes* are handled in exactly the same way.
- It is only possible to encode a *resource URI* by using a combination of the DC Identifier *property* and the DCTERMS URI *encoding scheme*. Thus, it is not possible to encode the *resource URI* in a simple DC record encoded using the XML encoding guidelines.

Appendix D - The abstract model and XHTML

This appendix compares the DCMI abstract model with the Expressing Dublin Core in HTML/XHTML meta and link elements DCMI recommendation.

Simple DC

Figure 11

Figure 11 shows an example simple DC description encoded according to the XHTML guidelines above. This example shows how the encoding supports the *property URI*, *value string*, *value string language* and *value URI* aspects of the DCMI abstract model. Again, it should be noted that the value of the DC Identifier *property* represented in this encoding syntax is denoted by a *value string*, even though it looks, to the human reader, as though it is a URI.

Qualified DC

Figure 12

Figure 12 shows an example qualified DC description encoded according to the XHTML guidelines above. This example shows how the encoding supports the *property URI*, *value string*, *value string language*, *value URI*, *encoding scheme URI*, *resource URI* and *resource class* aspects of the DCMI abstract model. Again, note that the 'dcterms:URI' *encoding scheme* is used to indicate that the content of the XHTML <meta> element is a *value URI* and that, although the *resource class* is indicated, the *class URI* is not encoded anywhere in this description.

Summary

The following aspects of the DCMI abstract model are supported by the Expressing Dublin Core in HTML/XHTML meta and link elements DCMI recommendation:

- *properties*
- *property URIs*
- *value strings*
- *value string languages*
- *value URIs*
- *encoding schemes*
- *encoding scheme URIs*
- *resource URIs*
- *resource classes*

The following aspects of the DCMI abstract model are not supported:

- *rich values*
- *related descriptions*
- *property*/sub-property relationships
- *resource class URIs*

The following constraints apply:

- Each *property* may have one *value string* (but not more than one) or a *value URI* but not both.
- *Vocabulary encoding schemes* and *syntax encoding schemes* are handled in exactly the same way.
- The *resource URI* may be implicit from the URI of the *resource* into which the *record* is embedded. Alternatively, the *resource URI* can be explicitly encoded by using a combination of the DC Identifier *property* and the DCTERMS URI *encoding scheme*.

Metadata associated with this resource: http://dublincore.org/documents/abstract-model/index.shtml.rdf

Title: DCMI Grammatical Principles
Creator: DCMI Usage Board
Identifier: http://dublincore.org/usage/documents/2003/11/18/principles/
Latest version: http://dublincore.org/usage/documents/principles/
Replaces: http://dublincore.org/usage/documents/2003/02/07/principles/
Date modified: 2003-11-18
Description: This document describes the grammatical principles that govern the decisions of the Usage Board as the maintenance body for DCMI metadata semantics. See also a related document, "DCMI Usage Board Mission" [1], and the Dublin Core Metadata Glossary [8].

1. Scope of this grammar

This grammar presents the typology of DCMI metadata terms and describes the principles underlying their definition and use. As defined in the "Namespace Policy for the Dublin Core", a DCMI term is "a DCMI element, a DCMI qualifier or term from a DCMI-maintained controlled vocabulary." A DCMI namespace, in turn, is "a collection of DCMI terms" [2].

2. Elements and qualifiers

2.1. Elements

An Element is a property of a resource. As intended here, "properties" are attributes of resources -- characteristics that a resource may "have", such as a Title, Publisher, or Subject.

2.2. Qualifiers

"Qualifiers" is the generic heading traditionally used for terms now usually referred to specifically as Element Refinements or Encoding Schemes.

2.2.1. Element Refinements.

An Element Refinement is a property of a resource which

shares the meaning of a particular DCMI Element but with narrower semantics. In some application environments (notably HTML-based encodings), Element Refinements are used together with Elements in the manner of natural-language "qualifiers" (i.e., adjectives) [3]. However, since Element Refinements are properties of a resource (like Elements), Element Refinements can alternatively be used in metadata records independently of the properties they refine [9]. In DCMI practice, an Element Refinement refines just one parent DCMI property.

2.2.2. Encoding Schemes.
An Encoding Scheme provides contextual information or parsing rules that aid in the interpretation of a term value. Such contextual information may take the form of controlled vocabularies, formal notations, or parsing rules. If an Encoding Scheme is not understood by a client or agent, the value may still be useful to a human reader. There are two types of Encoding Scheme:

2.2.2.1. Vocabulary Encoding Schemes
Vocabulary Encoding Schemes indicate that the value is a term from a controlled vocabulary, such as the value "China - History" from the Library of Congress Subject Headings.

2.2.2.2. Syntax Encoding Schemes
Syntax Encoding Schemes indicate that the value is a string formatted in accordance with a formal notation, such as "2000-01-01" as the standard expression of a date.

2.3. Dumb-down Principle

The qualification of Dublin Core Elements 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 term 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 an Element.

2.4. Appropriate values

Best practice for a particular Element or Qualifier may vary by context. Definitions may provide some guidance; other information may be found in the Usage Guide [6].

3. Vocabulary Terms

The Usage Board maintains the DCMI Type Vocabulary [7] -- a general, cross-domain list of recommended terms that may be used as values for the Resource Type element to identify the genre of a resource. The member terms of the DCMI Type

Vocabulary are called Vocabulary Terms.

If a Vocabulary Term is hierarchically related to another Vocabulary Term, the relationship indicators "Broader Than" and "Narrower Than" are used reciprocally in their term declarations.

4. Application Profiles

In DCMI usage, an Application Profile is a declaration of which metadata terms an organization, information resource, application, or user community uses in its metadata [10].

REFERENCES

[1] http://dublincore.org/usage/documents/mission/
[2] http://dublincore.org/documents/2001/10/26/dcmi-namespace/
[3] http://www.ietf.org/rfc/rfc2731.txt
[4] http://www.ukoln.ac.uk/metadata/dcmi/dc-xml-guidelines/
[5] http://dublincore.org/documents/dcq-rdf-xml/
[6] http://dublincore.org/documents/usageguide/
[7] http://dublincore.org/usage/terms/dcmitype/
[8] http://dublincore.org/documents/2001/04/12/usageguide/glossary.shtml
[9] A shift from the former view to the latter is reflected in the names assigned by the Usage Board to Element Refinements, with a move away from adjective-like names such as "created" (approved in July 2000) towards noun-phrase-like names such as "dateCopyrighted" (approved in July 2002). One consequence of using Element Refinements independently of Elements is that information about relationships between them will reside outside of the metadata records in separate schemas that applications needing to perform operations such as dumb-down will need to consult.
[10] http://dublincore.org/usage/documents/profiles/

Topic: Guidelines for Dublin Core Application Profiles
Modified: 2004-02-25 15:44, Wednesday
Maintainer: Tom Baker
Latest version: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/docs-DCAPs/
See also: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/

Topic shepherd: Tom Baker

SUMMARY

The "Guidelines for Dublin Core Application Profiles", approved as CEN Workshop Agreement CWA14855, are posted at:

http://dublincore.org/usage/meetings/2004/03/cwa14855-20040210.pdf

After a discussion in the Advisory Board, Makx posted a draft agreement to be reached with CEN about maintenance responsibility and rights with regard to CWAs of relevance to DCMI:

http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/CACHE/DCMI-CEN-agreement.pdf

As of mid-February, it appears that CEN has accepted the proposed agreement and formal agreement should follow soon. According to this agreement, DCMI has the right to post and edit a copy of a CWA (designated as a draft copy). Recommendations for changes will be accepted and incorporated by CEN unless CEN members object.

We had some substantive discussion of the DCAP guidelines in the Seattle meeting on which my notes show the following:

-- The issue of identifying terms with URIs as opposed to URI references should be discussed further in Section 3 (see also Sections 4.1 and 5.6). The issue is how to handle qualified names and the potential for confusion with XML qualified names, and deal with any other restrictions that are applied to how element and term names are documented.

Roland will present one or two slides in Bath about this issue.

-- Section 5.9 "Documenting Unorthodox Practices" should be rewritten with language that is more neutral with respect to conformance to a data model. It was felt that APs could often be used to document legacy practices, so the guidelines should recommend that legacy practices be described as they were implemented, explaining if appropriate that the practices are no longer in use. Tom would appreciate some further discussion on what should

actually be intended here.

In Bath, it would be good if we could:

-- Agree in principle that CWA14855 should be recommended and
pointed to by the Usage Board as good-practice guidelines
for writing an application profile.

-- Identify (or confirm) which sections, other than the above,
may need revision, specifying how they should be revised.

-- Schedule a simple process for signing off on the revisions
over the list.

The goal would be to specify how the guidelines could be
designated a DCMI Recommended Resource.

Date: Tue, 10 Feb 2004 17:45:22 -0000
From: Pete Johnston <p.johnston@UKOLN.AC.UK>
Subject: Re: Bath agenda: CEN application profile guidelines
To: DC-USAGE@JISCMAIL.AC.UK

> -- The issue of identifying terms with URIs as opposed to
> URI references should be discussed further in Section 3
> (see also Sections 4.1 and 5.6). The issue is how to handle
> qualified names and the potential for confusion with XML
> qualified names, and deal with any other restrictions that
> are applied to how element and term names are documented.
> I would appreciate if someone more knowledgeable on the
> state of discussion (Roland?) would volunteer to explain
> the issue here.

At the risk of butting in ;-) I think the "MODS location"
issue is directly relevant here too.

Leaving aside the URI v QName question for a moment, the CEN
DCAP guidelines document talks about "metadata terms" and
"elements from other vocabularies", which can then be "used"
in a DCAP.

I wonder whether there is an element of assumption here that
all metadata vocabularies are being developed within the same
conceptual framework as Dublin Core, and that the descriptions
of those other vocabularies will use terminology like "term",
"element" and "refinement" in the same way as DCMI does. Or
at least that those DC concepts can be projected or mapped
very easily onto concepts within the models and frameworks
used by other metadata standards.

Now that may well be the case, particularly where those "other vocabularies" have been developed with interoperability with DC in mind - but I'd argue that it is not always the case.

And I worry that we risk encouraging people to see what my French teacher used to call "false friends" - like the "DC element"/XML element case with MODS. It seems to me that the problem is not only one of what sort of identifier is used, but what are the things that are being identified and referenced, and whether they are appropriate for (re)use in a DCAP at all. And to provide the basis for those decisions, we need a model for what a DCAP _is_, which I'm not really sure we have at the moment.

Date: Wed, 11 Feb 2004 12:51:57 +0100
From: Roland Schwaenzl <Roland.Schwaenzl@MATHEMATIK.UNI-OSNABRUECK.DE>
Subject: Re: Bath agenda: CEN application profile guidelines
To: DC-USAGE@JISCMAIL.AC.UK

> I would appreciate if someone more knowledgeable on the
> state of discussion (Roland?) would volunteer to explain
> the issue here.

Glad to do so. Do you want this by mail or at Bath?
By the way RDF and OWL are now W3C Recommendations.

Date: 2003-11-21
From: Makx Dekkers
Posted to dc-ab@jiscmail.ac.uk

Dear all,

Following our discussions in Seattle concerning the potential relation with CEN, I have worked with the people at CEN to draft an agreement.

For those of you who haven't followed the discussion in detail, CEN is a European standardisation organisation (http://www.cenorm.be/), funded by national standard bodies in Europe and by the European Commission.

The workshop that is relevant for DCMI is the Workshop on Dublin Core Metadata: www.cenorm.be/sh/mmi-dc/. CEN Workshops are very much like DCMI Working Groups: they work through open mailing lists, open FTP archives, and open meetings, and have a strictly consensus-based approach. Results of these workshops are called CWA, CEN Workshop

Agreements, which are not formal standards but rather documented agreements within a particular workshop.

Through the workshop mentioned above, CEN is willing to fund activities related to Dublin Core, mostly production of DC-related documentation and guidelines. Both CEN and DCMI are interested to clarify and formalise the relationship between the two organisations in terms of sharing responsibility and credit for developing specifications and advancing DCMI-related work items.

The draft agreement below, on one hand, takes into account the requirements of DCMI as expressed in the discussion in the UB and AB meetings in Seattle, and, on the other hand, respects the policies of CEN as an organisation funding documentation work that is useful and relevant to DCMI.

The underlying idea is that it is beneficial to DCMI and the DCMI community that a funding organisation like CEN is willing and able to channel funds into work that is useful to DCMI.

In preparing the text, we have used a number of principles based on general DCMI principles and on the comments from UB and AB in Seattle.

My summary of these guiding principles:

- Respect DCMI general principles regarding open and unrestricted participation and free availability of results
- Allow direct involvement from DCMI and the DCMI community in the development and maintenance process
- Allow for assignment of official DCMI status to results
- Allow DCMI to submit these results for further national and international standardisation
- Safeguard the possibility to update, modify or deprecate the results as necessary
- Allow DCMI to assume stewardship and control over results if third-party involvement cannot be sustained

Following the text, I have included my analysis of how this text addresses the worries that were expressed in Seattle.

This is still draft text. It is now being submitted to the CEN Board for approval at the same time that I, as promised in Seattle, share it with the DCMI Advisory Board.

I would like you to critically review the text and send your comments to the list.

DRAFT CEN-DCMI AGREEMENT FOR REVIEW BY DCMI ADVISORY BOARD

1. CEN will make available the CWAs that are produced by its Workshop MMI-DC for download on the web, under the usual CEN conditions for download (for personal use only, etc). DCMI may include a link to the CWA at the CEN Website

2. DCMI may, but is not obliged to, assign an official DCMI status to documents produced and agreed by CEN Workshops, specifically DCMI Recommendation or DCMI Recommended Resource. DCMI will clearly indicate that such documents have been produced in the context of CEN. Availability of these documents will be through linking to the CWAs concerned on the CEN web-site.

3. CEN grants a royalty-free copyright licence to DCMI to modify CEN CWAs, with full acknowledgement of the original source, and to put these modified ones on the DCMI Web site

4. DCMI commits to provide feedback to CEN on the CWA so that the CWA can be updated through CEN procedures. CEN will take the necessary steps, through its Workshop procedure, to update its CWAs, when requested by DCMI.

5. During development of a CWA, CEN grants DCMI the right to post working drafts on the DCMI Web site and invite public comment and contributions, with a clear indication that this work takes place in the context of CEN. DCMI will channel comments from the DCMI community to the CEN Workshop.

6. In such cases as CEN were no longer able to maintain a Dublin-Core-related CWA, DCMI may, but is not obliged to, take over the maintenance and further development of such document under the normal rules and procedures governing work within DCMI, acknowledging that the original version of the document was produced in the context of CEN and referring to the latest official CWA version.

7. DCMI may, in consultation with CEN, propose a CWA - prepared in the context of CEN and endorsed as a DCMI Recommendation - for consideration by global standards organizations such as ISO and IETF, for approval as an international standard, Internet Standard or RFC.

DISCUSSION

As far as I understand, the explicit and implicit concerns expressed in Seattle were (excuse blunt paraphrasing):

1. The work in CEN leads to documents that are wrong in some way, e.g. are contrary to DCMI principles or policies.

Clauses 4 and 5 intend to ensure that DCMI can contribute to the content and quality of the work so that we have a possibility to influence the CEN results.

In the situation that, notwithstanding the contributions from the DCMI community, the end result is below our standards, Clause 2 makes it possible that we do /not/ adopt the CWA as a DCMI Recommendation.

2. The work in CEN does deliver a good and appropriate document but DCMI cannot use or distribute it.

   This concern is covered in Clauses 1 and 3.

3. After publication of a CWA, CEN or others may run away with it and disallow DCMI to further develop the document.

   Clause 3 gives us the right to further develop any documents that are produced through CEN funding, without restriction.

4. After CEN ceases to be interested, the CWA is dead and we have no right to further develop the document.

   This is addressed by Clause 6, but note that this does not put us under obligation to take over maintenance after the work in CEN ceases.

5. A good CWA that DCMI endorses cannot be promoted to international standard.

   Clause 7 covers this, with the additional remark that CEN would be happy if this occurred.

6. Developing documentation in a regional organisation would be in conflict with the global nature of DCMI

   CEN rules and procedures do allow full and unrestricted participation of non-Europeans in their Workshops

CEN

WORKSHOP

AGREEMENT

CWA 14855

November 2003

ICS 35.100.05; 35.240.60

English version

Dublin Core Application Profile Guidelines

This CEN Workshop Agreement has been drafted and approved by a Workshop of representatives of interested parties, the constitution of which is indicated in the foreword of this Workshop Agreement.

The formal process followed by the Workshop in the development of this Workshop Agreement has been endorsed by the National Members of CEN but neither the National Members of CEN nor the CEN Management Centre can be held accountable for the technical content of this CEN Workshop Agreement or possible conflicts with standards or legislation.

This CEN Workshop Agreement can in no way be held as being an official standard developed by CEN and its Members.

This CEN Workshop Agreement is publicly available as a reference document from the CEN Members National Standard Bodies.

CEN members are the national standards bodies of Austria, Belgium, Czech Republic, Denmark, Finland, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Luxembourg, Malta, Netherlands, Norway, Portugal, Slovakia, Spain, Sweden, Switzerland and United Kingdom.

EUROPEAN COMMITTEE FOR STANDARDIZATION
COMITÉ EUROPÉEN DE NORMALISATION
EUROPÄISCHES KOMITEE FÜR NORMUNG

Management Centre: rue de Stassart, 36 B-1050 Brussels

Ref. No. CWA 14855:2003 D/E/F

Contents

Foreword

The production of this CEN Workshop Agreement (CWA) was formally accepted as part of the CEN/ISSS Workshop on Metadata for Multimedia Information - Dublin Core (WS/MMI-DC) in the Workshop's plenary meeting on 2002-03-07.

This CWA was agreed upon by the contributing partners in the CEN/ISSS Workshop on MMI-DC, representing a wide mix of interests, including administrations, libraries, on-line education and geographic information systems. The list of company individuals who have supported the document's contents may be obtained from the CEN/ISSS Secretariat.

The CWA was approved by the Workshop’s plenary meeting on 2003-09-08.

The final text of this CWA was submitted to CEN for publication on 2003-09-09.

Introduction

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.

A DCAP is designed to promote interoperability within the constraints of the Dublin Core model and to encourage harmonization of usage and convergence on “emerging semantics" around its edges. Historically, application profiles have emerged out of a need to share local domain- or application-specific refinements of or extensions to Dublin Core within particular application communities without necessarily seeking an extension of the core standard maintained by the Dublin Core Metadata Initiative (DCMI). Application profiles document how implementers use elements from Dublin Core along with elements from other vocabularies, customizing standard definitions and usage guidelines for local requirements [HEERY].

In practice, application profiles are created for a wide range of purposes: to document the semantics and constraints used for a set of metadata records (“instance metadata”); to help communities of implementers harmonize metadata practice among themselves; to identify emerging semantics as possible candidates for formal standardization; as guides for semantic crosswalks and format conversions; as specifications for formal encoding structures such as Document Type Definitions (DTDs); for interpreting or presenting legacy or proprietary metadata in terms of widely-understood standards; or for documenting the rules and criteria according to which a set of metadata was created. Application profiles often represent “work in progress”, providing foci for ongoing efforts to incrementally improve and clarify a body of shared metadata semantics within a particular user community.

In the absence of guidelines, creators of application profiles have hitherto invented a wide range of presentation formats. The present document distils the salient features of many existing profiles into a format that is as concise and simple as possible, yet as precise and detailed as is sometimes necessary to support the various uses identified above.

Semantic interoperability – the ultimate purpose of documents such as DCAPs – is a longer-term goal to be pursued as metadata vocabularies and related enabling technologies mature over time. In their current form, DCAPs are designed to document metadata usage in a normalized form that will lend itself to translation into common models, such as RDF, that can be processed by machines to automate such interoperability.

Machine-understandable representations will achieve this goal to the extent that metadata terms can be referenced using stable, well-documented identifiers. As discussed below, the practice of identifying metadata terms with Uniform Resource Identifiers (URIs) is currently gaining momentum. Maintaining a DCAP over time, then, may involve improving its precision incrementally by identifying its terms with URIs as the URIs become available; this is referred to here as the Principle of Appropriate Identification.

In the meantime, these guidelines aim at the more modest aim of providing system developers and information specialists with a normalized and readable view of Dublin-Core-based metadata models. A DCAP should include enough information to be of optimal usefulness for its intended audience – a Principle of Readability – even if this entails the redundant inclusion of information, which, in a formal system of machine-processable schemas, might otherwise be fetched dynamically from external sources.

Given the flexibility of presentation required by the Principle of Readability, no assumption is made that DCAPs will be convertible into future machine-understandable forms without the use of ad-hoc heuristics or manual intervention. Creators of DCAPs should bear in mind that a normalized form of documentation cannot itself address the deeper problems of interoperability in a world with a diversity of underlying metadata models – problems which will continue to challenge the metadata community as a whole, and the Dublin Core Metadata Initiative in particular, for the foreseeable future.

1 Scope

The present document gives guidance on how information should be structured and presented in Dublin Core Application Profiles. Principles and concepts underlying DCAPs as declarative metadata constructs are defined and explained.

The guidelines do not mandate a particular document format for DCAPs. DCAPs may be presented as plain text files or as Web pages, word-processing files, PowerPoint, or indeed as ink on paper. By providing a consistent presentation structure for such documents, these guidelines aim at making it easier for people to understand what others are doing in their metadata. The guidelines mandate enough structure to ensure that DCAPs will be convertible as straightforwardly as possible into expressions that use schema languages, such as RDF, for automatic processing by machines. In this sense, a normalized documentational form for DCAPs is a first step towards the more ambitious and long-term goal of automating semantic interoperability across a broad diversity of information sources.

2 Definitions

Dublin Core Application Profile (DCAP): A DCAP is a declaration specifying which metadata terms an organization, information provider, or user community uses in its metadata and how those terms have been customized or adapted to a particular application. By definition, a DCAP is based in part on Dublin Core and follows DCMI Grammatical Principles [DCMI-PRINCIPLES]. A DCAP consists of a Descriptive Header and one or more Term Usages.

DCMI Grammatical Principles: As maintained by the Dublin Core Metadata Initiative, DCMI grammatical principles specify a typology of metadata terms – Elements, Element Refinements, Encoding Schemes, and Vocabulary Terms – along with their interrelationships and functions [DCMI-PRINCIPLES]. A DCAP is based on the simple model of a resource described with a flat set of properties. This is consistent with DCMI grammatical principles, which do not themselves specify more elaborate models.

Descriptive Header: A Descriptive Header places the DCAP into an interpretive context by specifying, at a minimum, a Title, Creator, Date, Identifier, and Description for the DCAP. An optional Preamble may comment on any technical or stylistic conventions followed in the DCAP.

Term Usage: A Term Usage is a description of a metadata term, which, at a minimum, identifies a metadata term in accordance with the Principle of Appropriate Identification by using one or more identifying attributes – Term URI, Defined By, Name, Label – as described in Section 3. Optionally, a Term Usage may also describe or annotate a term in more detail by providing additional definitional attributes, relational attributes, or constraints, as described in Section 4.

Principle of Appropriate Identification: The Principle of Appropriate Identification dictates that metadata terms be identified as precisely as possible. As established in the so-called CORES Resolution of December 2002, the preferred method for identifying a metadata term is to cite its Uniform Resource Identifier (URI). All DCMI metadata terms are identified with URIs, and URIs are currently being assigned to the terms of other major semantic standards such as MARC21 and IEEE/LOM [CORES-RESOLUTION]. Whenever such URIs are available they should be cited as an attribute of a Term Usage (its “Term URI”). Terms to which URIs have not (or not yet) been assigned should be identified using other attributes as appropriate, as described in Section 3.

Principle of Readability: The Principle of Readability dictates that a DCAP should include enough information in Term Usages to be of optimal usefulness for the intended audience of the DCAP – even if this entails the redundant inclusion of information which, in a formal system of machine-processable schemas, might otherwise be fetched dynamically from external sources. Conversely, the Principle of Readability allows unused attributes simply to be omitted from display.

3 Identifying terms with appropriate precision

Application profiles serve to clarify who is declaring and maintaining the metadata semantics that a group wants to share. This section describes how a metadata term used in a Term Usage can be identified with appropriate precision (the Principle of Appropriate Identification).

At present, the preferred method for identifying a metadata term is to cite its Uniform Resource Identifier (URI) if such is available. A URI is "a compact string of characters for identifying an abstract or physical resource" constructed according to a generic and flexible syntax [URI]. The World Wide Web Consortium has promoted the notion that “All important resources should be identified by a URI” [WEBARCH] and has specifically promoted the use of URIs for identifying metadata elements. In the CORES Resolution of December 2002, the maintainers of seven leading metadata standards – Dublin Core, IEEE/LOM, DOI, CERIF, MARC21, ONIX, and GILS -- pledged to assign URIs to their elements and to articulate policies for the persistence of those URIs [CORES-RESOLUTION]. (Note that a URI, when used to identify a metadata term, often functions as a Web address for accessing information about that term, such as a Web page or machine-processable schema. However, the CORES Resolution does not require that such identifiers resolve to such resources, and URIs that result in “file not found” messages are not necessarily “broken” as identifiers.)

For metadata terms to which a URI has been officially assigned – for example, by DCMI or by another signatory of the CORES Resolution – that URI should be cited in the field “Term URI”. For example, the Dublin Core element “Audience” should be cited as “http://purl.org/dc/terms/audience”. As this form of identification is precise and sufficient on its own, other identifying fields may be left blank:

```
Term URI        http://purl.org/dc/terms/audience
Name            -
Label           -
Defined By      -
```

In accordance with the Principle of Readability, other identifying attributes such as Name and Label could be added here to make the DCAP more “reader-friendly”. If the DCAP is intended as a guide for processing metadata records, it may indeed be necessary to provide a Name (i.e., the string actually used in the metadata records). If the Name or Label of a term are considered more “reader-friendly” as captions for a Term Usage than the Term URI, the order of these attributes may be changed to put these first. See Sections 5.4 (“Attributes copied from external sources”) and 5.2 (“Readability of Term Usages”) for further discussion.

A term that has been declared or documented somewhere but not assigned a URI (as far as one knows) should be identified as precisely as possible by providing its name and pointing to a declarative document or schema in which it has been defined. The declarative document or schema should be cited with URI, Web address, or bibliographic reference in the field "Defined By". The term itself can be cited using either a string identifier or token (in the field “Name”, which by default is assumed to be case-sensitive) or a natural-language label (in the field "Label"), or both, taken from the declarative document or schema:

```
Term URI        -
Name            AttendancePattern
Label           Attendance Pattern
Defined By      http://someones-project.org/schema.html
```

For a term that has not already been defined in any other declarative document, the field Defined By should simply cite the URI of the DCAP itself (as assigned with Identifier in the DCAP Descriptive Header). For example, in a DCAP with the URI "http://my-project.org/profile.html", a new local term called Star Ratings could be defined as follows:

```
Term URI        -
Name            StarRatings
Label           Star Ratings
Defined By      http://my-project.org/profile.html
```

A creator of a DCAP wishing to declare locally coined terms in a way that makes them citable with precision, and thus re-usable by others, may undertake the additional step of assigning them URIs. At present, the technical conventions and "Web etiquette" for naming metadata terms with URIs have yet to establish themselves in common practice, though at a minimum it seems both polite and sensible not to promote new URIs unless it is expected they will be maintained. For the purposes of DCAPs, DCMI itself provides models of practice, and further options are likely to emerge as the CORES Resolution is implemented [DCMI-NAMESPACE, DCMI-TERMS, DCMI-SCHEMAS]. Note that the CORES Resolution itself addresses the use of URIs as identifiers only and is silent on whether the URIs should resolve to informational Web pages or schemas [CORES-RESOLUTION].

4 Attributes of a Term Usage

Attributes for describing the metadata terms "used" in a DCAP are listed below. Note that they are called "attributes" here simply to avoid confusingly recursive formulations such as "terms for describing terms".

Use of Identifying Attributes in Term Usages (see Section 4.1) is governed by the Principle of Appropriate Identification. According to this principle, a Term Usage should use one or more of the four Identifying Attributes to identify a term as precisely as appropriate – i.e., with a formally assigned URI if available, or alternatively by citing a name or label for the term along with a reference to a document, schema, or Web page in which that term is defined.

All of the other attributes of Term Usages are optional and should be used as local needs may dictate. As discussed in Section 5.4, “local” and “source” attributes may be distinguished as necessary.

4.1 Identifying attributes

Term URI	A Uniform Resource Identifier used to identify the term.
Name	A unique token assigned to 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.

4.2 Definitional attributes

Definition	A statement that represents the concept and essential nature of the term.
Comments	Additional information about the term or its application.
Type of term	A grammatical category of the term (e.g., "Element", "Element Refinement", or "Encoding Scheme").

4.3 Relational attributes

Refines	The described term semantically refines the referenced term.
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.
Has Encoding Scheme	The described term is qualified by the referenced Encoding Scheme.
Similar To	The described term has a meaning the same as, or similar to, that of the referenced term.

4.4 Constraints

Obligation	Indicates whether the element is required to always or sometimes be present (i.e., contain a value). Examples include "Mandatory", "Conditional", and "Optional".)
Condition	Describes the condition or conditions according to which a value shall be present.
Datatype	Indicates the type of data that can be represented in the value of the element.
Occurrence	Indicates any limit to the repeatability of the element.

5 Discussion

5.1 Descriptive Headers

By definition, a DCAP consists of a Descriptive Header and one or more Term Usages (see Section 2). The Description Header should include the following:

- A brief description of the DCAP based on Dublin Core. At a minimum, the description should specify a Title, Contributor, Date, Identifier, and Description, as explained in more detail in Annex A. Ideally, the description of the DCAP will elaborate on the context in which the DCAP is intended to be used.

- Optionally, a Preamble for the DCAP should describe any technical or formatting conventions used in the DCAP. For example, if namespace prefixes are used in the Name field (see Section 5.6), these prefixes should be documented here. The Preamble can also cite Web pages or schemas in which the terms used in the DCAP are documented and defined so that such information does not need to be repeated in the "Defined By" field of each individual Term Usage; see the example in Section 6.1.1.

5.2 Readability of Term Usages

By default, each term cited in a DCAP should be described with its own Term Usage – a table with a full set of attributes on the left and attribute values on the right. In accordance with the Principle of Readability, however, the intended use of a DCAP may dictate a different presentational style: while DCAPs intended for use by software developers will need to be explicit and detailed, DCAPs intended primarily as informational documents for human consumption can (and often should) be much terser. The following are several ways in which Term Usages may be formatted for readability:

- Instead of creating a separate Term Usage for every Element Refinement and Encoding Scheme used in an application, such terms may simply be cited in the attributes Refined By or Has Encoding Scheme of the Term Usages of the Elements to which they refer (see Section 5.3). Note that this terser style does not support the addition of usage notes, local definitions, or annotations, for which a full Term Usage must be used.

- Attributes not needed for Term Usages can simply be omitted. At one extreme, a Term Usage might legitimately consist of just a Name and a Term URI; see the example in Section 6.1.2).

- The order of attributes presented in Section 4 is significant only for the usability of DCAPs as documents – not for future machine-processable representations of DCAPs. Authors of DCAPs may therefore change the order of attributes in the interest of readability, though they should bear in mind that any such changes may make it more difficult for people to compare two DCAPs visually. For examples of how Name (in boldface) is placed before Term URI, see Section 6.1.2 and DCMI-TERMS.

- In the interest of readability, it might make sense to describe Elements, Element Refinements, and Encoding Schemes with different subsets of relevant attributes. Indeed, these different types of terms might be grouped under separate sections of the DCAP document. (For example, see how Elements and Element Refinements are separated from Encoding Schemes in the document "DCMI Metadata Terms" [DCMI-TERMS].)

- In the interest of readability and of future machine-parsability, attributes should be repeated when necessary (as opposed to listing multiple values for a single attribute).

5.3 "Using" Element Refinements, Encoding Schemes, and Vocabulary Terms

5.3.1 Using Vocabulary Terms

According to DCMI Grammatical Principles, a Vocabulary Term is a member of a controlled vocabulary of values, and a controlled vocabulary of values (as a whole) is named by an Encoding Scheme [DCMI-PRINCIPLES].

In general, it is not the role of application profiles to declare controlled vocabularies of values, either in the sense of creating lists of potential values or in the sense of giving that list (as a whole) a name and URI. Sets of Vocabulary Terms are most appropriately declared in separately citable documents external to a DCAP.

However, if the creator of a DCAP merely wishes to specify a short list of possible values (e.g., "Animal, Vegetable, or Mineral"), these can be simply listed in a "Comment" field.

5.3.2 Using Encoding Schemes

There are three ways to cite Encoding Schemes in a DCAP:

- The most concise way is to use the attribute Has Encoding Scheme (or Comment) for a blanket reference to a set of encoding schemes documented elsewhere. The DCAP for Resource Discovery Network, for example, simply cites "RDN Subject Encoding Schemes" and gives a URL where the list of those encoding schemes may be found; see the example in Section 6.1.2.

- A more precise way is to use the attribute Has Encoding Scheme, repeated as necessary, to cite each Encoding Scheme by its URI (or by Name and URI); for example, see the example in Section 6.3.2.

- In addition to citing Encoding Schemes in the Has Encoding Scheme attribute of Elements, creators of DCAPs may want to describe Encoding Schemes in stand-alone Term Usages in order to annotate their usage, for example by specifying a Datatype, Occurrence, or Local Definition. The attribute Encoding Scheme For points back to the Element or Element Refinement qualified.

5.3.3 Using Element Refinements

The options for Element Refinements are analogous to those for Encoding Schemes:

- Statements such as "all terms in Vocabulary D can be used as element refinements for Contributor" can be simply recorded in a Refined By attribute (or as a Comment).

- Element Refinements can be cited one-by-one using the attribute Refined By; see the example in Section 6.3.2.

- Element Refinements can additionally be described in separate Term Usages.

5.4 Attributes copied from external sources

Ideally, application profiles would be dynamically up-dated with information on the terms they use directly from schemas on the Web and this information would be integrated with local annotations into a "one-stop" document for the convenience of users. The use of machine-understandable DCAPs may some day make this possible.

In the meantime, however, creators of DCAPs who wish to include definitions or other such information from original source documents in their Term Usages have no choice but to copy that information from the source. While the Principle of Readability specifically permits this, authors of DCAPs should bear in mind that copied information, if not maintained, can go out of alignment with the official source.

Where information copied from external sources is supplied, this fact should be reported in the Preamble as described in Section 5.1. Where it is necessary to distinguish in a DCAP between attributes defined locally and attributes copied from an external source, the DCAP should establish its own document-internal

convention, such as distinguishing between a Local Definition and a Source Definition; see the example in Section 6.3.2.

5.5 Types of Comments

Past creators of application profiles for Dublin Core have invented many types of annotation, the most popular of which have been Notes, Best Practice, Usage, Scope, Open Questions, Examples, Purpose, Guidelines, and Don't Confuse With. While the present guidelines lump all of the above into a generically named Comments field, creators of DCAPs may wish to repeat this field with different labels as needed. The needs of future machine processing do not now seem to dictate tighter uniformity in this area.

5.6 Term URIs versus Qualified Names

In the sense intended here, Qualified Names are names of metadata terms that are “qualified” with a prefix standing for a namespace with which the terms are associated (a “namespace prefix”). For example, the Dublin Core element “Title” is sometimes referenced in metadata records and usage documentation using a namespace prefix such as “DC.” or “dc:” as in “DC.Title” or “dc:title”. As straightforward as this citation method may seem, it is based on assumptions about the nature of “namespace” that cannot be assumed to hold across different application environments (e.g., HTML versus RDF versus relational databases) or metadata communities (e.g., for citing elements from standards other than Dublin Core), and at any rate it presupposes an additional mechanism or declaration for associating prefixes with the proper namespaces.

For such reasons, it is far better to cite an element with a full URI – indeed, this is the only method supported by the CORES Resolution and by DCMI policy [CORES-RESOLUTION, DCMI-NAMESPACE]. According to the Principle of Appropriate Identification followed in these guidelines, a Term URI must be cited when available.

On the other hand, long strings such as “http://purl.org/dc/elements/1.1/title” are not very readable and may be misunderstood by the average reader of a DCAP. In accordance with the Principle of Readability, therefore, the author of a DCAP may choose to use qualified names (e.g., “dc:title”) in the “Name” field – as long as any prefixes used are explained in the Preamble of the DCAP, and as long as any available Term URIs are cited as well.

5.7 Declaring new elements

There is nothing to restrain the creator of a DCAP from creating new URIs as identifiers for locally coined metadata terms. For reasons discussed above in Section 3, one should perhaps pause for reflection before taking this step, and if URIs are declared, this step should perhaps be documented separately and not embedded “in passing” into a DCAP full of Term Usages. Any URIs declared for use in a DCAP might best be formed by following the DCMI algorithm and concatenating the URL of the DCAP (e.g., "http://myproject.org/profile/") and the Name of the term (e.g., "starRatings") into a single string (e.g., "http://myproject.org/profile/starRatings") [DCMI-NAMESPACE]. Other models for forming URIs as identifiers for metadata elements are emerging with the implementation of the CORES Resolution [CORES-RESOLUTION].

5.8 Documenting grouped or nested metadata elements

In order to be usable across a diversity of application environments, Dublin Core was designed as a flat set of attributes for describing a resource. In implementation practice, however, Dublin Core elements may be embedded in more elaborate models that group or nest the elements in locally specific ways.

In the absence of a clear and widely accepted data model beyond that of the flat set of attributes, however, applications for integrating metadata from many different sources may be able only to extract and interpret the metadata in terms of Simple Dublin Core, losing any application-specific modelling context. An application designer wishing to document nesting or grouping constructs in a DCAP will need to extend the guidelines described here in order to do so and should bear in mind that documenting such constructs will not in itself guarantee that they will be understood or correctly processed by other applications.

5.9 Documenting unorthodox practices

For reasons both of history and of expedience, a significant number of applications have metadata based on interpretations of the Dublin Core model that are unsound from the standpoint of today's grammatical principles. For example, an application may use CreatorDateOfBirth – an element representing the birth date of a creator of a resource that does not, however, semantically "refine" Creator as its name may imply.

Rather than incorrectly asserting "CreatorDateOfBirth" to be an Element Refinement refining http://purl.org/dc/elements/1.1/creator, the Term Usage in the DCAP should simply record the local name of the element and identify the URI of the DCAP itself as its source. For example, if the DCAP itself is identified by "http://myproject.org/profile/2003/03/17/", the Term Usage should declare the following, leaving empty any fields (such as "Term URI" and "Refines") that would make incorrect assertions about the element:

```
Term URI        -
Local Name      CreatorDateOfBirth
Defined By      http://my-project.org/profile.html
Refines         -
```

Whether "errors" such as "CreatorDateOfBirth" will be of negative consequence for interoperability will depend on how they are interpreted and used in the context of particular applications. The analytical effort involved in creating a DCAP is in effect an important first step towards putting such applications onto a more interoperable foundation.

6 Examples

6.1 UK Resource Discovery Network OAI Application Profile

6.1.1 Descriptive Header

Title	RDN OAI Application Profile
Contributor	Andy Powell
Date	2003-03-23
Identifier	URL for this document – to be assigned
Description	This document expresses the application profile established by the Resource Discovery Network (RDN) to be used by RDN partners for harvesting of records using the Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH). The Application Profile is expressed according to guidelines published by the CEN/ISSS [Reference]. Full user documentation for the Application Profile, together with associated XML schemas, is available at http://www.rdn.ac.uk/oai/rdn_dc/. All Dublin Core terms are fully documented at http://www.dublincore.org/documents/dcmi-terms/.

6.1.2 A Term Usage

Name	Subject
Term URI	http://purl.org/dc/elements/1.1/subject
Has Encoding Scheme	DC Subject Encoding Schemes
Has Encoding Scheme	RDN Subject Encoding Schemes
Comment	RDN Subject Encoding Schemes are available from http://www.rdn.ac.uk/publications/cat-guide/subject-schemes/
Obligation	Recommended

6.1.3 Commentary

The DCAP for the UK Resource Discovery Network is formatted in the tersest possible style [RDN]. Note in particular the following:

- The Descriptive Header puts the DCAP into a specific usage context.
- Only the attributes actually used in a given Term Usage are shown. Indeed, most of the Term Usages in this DCAP consist of just a Name and a Term URI.
- Instead of listing all of the DCMI- and RDN-maintained Encoding Schemes as separate Has Encoding Scheme entries (or as separate Term Usages), this DCAP uses shorthand references to “DC Subject Encoding Schemes” and “RDN Subject Encoding Schemes”. Pointers to documentation are given in the Descriptive Header (for the former) and in a Comment field of the Term Usage (for the latter).
- This DCAP increases the readability of Term Usages by listing the Name, in boldface, before the Term URI. This option is discussed in Section 5.2.

6.2 Renardus Application Profile

6.2.1 Descriptive Header

Title	Renardus Application Profile
Contributor	Metadata Working Group SUB Göttingen
Date	18-04-2002
Identifier	http://renardus.sub.uni-goettingen.de/renap/renap.html
Description	Cross search and cross browse European quality controlled subject gateways.

6.2.2 A Term Usage

Term URI	http://purl.org/dc/elements/1.1/language
Name	Language
Label	Language
Defined By	-
Definition	-
Comments	Renardus: The language code is the ISO 639-2, three-letter code. SUB will provide a mapping between the two letter and three letter language code but this will also be found on the LoC site – ISO 639-2: http://lcweb.loc.gov/standards/iso639-2/englangn.html
Type of term	Element
Refines	-
Refined By	-
Encoding Scheme For	-
Has Encoding Scheme	http://purl.org/dc/terms/ISO639-2
Similar To	-
Obligation	Mandatory
Condition	-
Datatype	String
Occurrence	Repeatable

6.2.3 Commentary

The DCAP for the Renardus Project has been formatted in a somewhat more verbose style [RENARDUS]. Note in particular:

- The DCAP uses its own URL as an identifier.

6.3 UK e-Government Metadata Standard Application Profile

6.3.1 Descriptive Header

Addressee	Metadata Working Group, Interoperability Working Group
Contributor	Drafted by Interoperability and Metadata Analyst, Office of the e-Envoy, Cabinet Office, UK farah.ahmed@e-envoy.gsi.gov.uk
Contributor	Metadata Working Group
Coverage.spatial	UK
Creator	Senior Policy Advisor, Interoperability and Metadata, Office of the e-Envoy, Cabinet Office, UK
Date.issued	2003-08-05
Description	The elements and refinements that provide the structure for metadata used by the UK public sector, designed to complement the e-GMS.
Format	Text/MS Word 2003
Identifier	http://purl.oclc.org/NET/e-GMS-AP_v1
Language	Eng
Publisher	Office of the e-Envoy, Cabinet Office, UK. govtalk@e-envoy.gsi.gov.uk
Rights.copyright	http://www.hmso.gov.uk/docs/copynote.htm Crown Copyright
Source	http://purl.oclc.org/NET/e-GMS_v2
Status	Version 1.0 For publication
Subject	Metadata
Subject.category	Information management
Title	UK e-government metadata standard application profile version 1

6.3.2 A Term Usage

Term URI	http://purl.org/dc/elements/1.1/date
Defined by	http://purl.org/dc/elements/1.1/
Name	Date
Label	Date
Source Definition	A date associated with an event in the life cycle of the resource.
Source Comment	-
Local Definition	-
Local Comment: Purpose	To enable the user to find the resource by limiting the number of search hits according to a date, e.g. the date the resource was made available.
Local Comment: Notes	Dates need to appear in a format that is recognisable to people all over the world, and that can be interpreted by computer software. The W3C format allows accurate searching, and makes it clear which is the year, month or day. The format is ‘ccyy-mm-dd’, where ‘ccyy’ is the year, ‘mm’ is the month and ‘dd’ the day. When the time is also needed, add ‘hh:mm’, where ‘hh’ is the hour (using the 24 hour clock), ‘mm’ is minutes. More about this notation can be found at http://www.w3.org/TR/NOTE-datetime.

Local Comment: Not to be confused with	Coverage – Date refers to dates relevant to the information resource itself, not the information held within the resource. For example, for a document about the civil service in the 18th century, put '18th century' in Coverage and put the date published in Date.
Type of term	Element
Refines	-
Refined by	http://www.govtalk.gov.uk/terms/dateAcquired
Refined by	http://purl.org/dc/terms/dateAccepted
Refined by	http://purl.org/dc/terms/Available
Refined by	http://purl.org/dc/terms/dateCopyrighted
Refined by	http://purl.org/dc/terms/created
Refined by	http://purl.org/dc/terms/issued
Refined by	http://purl.org/dc/terms/dateSubmitted
Refined by	http://purl.org/dc/terms/valid
Refined by	http://purl.org/dc/terms/modified
Refined by	http://www.govtalk.gov.uk/terms/cutOffDate
Refined by	http://www.govtalk.gov.uk/terms/dateDeclared
Refined by	http://www.govtalk.gov.uk/terms/dateClosed
Refined by	http://www.govtalk.gov.uk/terms/nextVersionDue
Refined by	http://www.govtalk.gov.uk/terms/updatingFrequency
Encoding Scheme For	-
Has Encoding Scheme	http://purl.org/dc/terms/W3CDTF
Has Encoding Scheme	http://purl.org/dc/terms/Point
Similar To	-
Constraints	The value must always be taken from the specified encoding scheme, with the exception of the 'updatingFrequency' refinement.
Obligation	Mandatory
Condition	A value must be given either for the unqualified date or at least one date refinement
Datatype	-

6.3.3 Commentary

The DCAP for the UK e-Government Metadata Standard is formatted in the most detailed and specific possible style [EGMS]. While this results in a significantly longer document than the DCAPs for RDN and Renardus, such specificity may be helpful to developers of applications that need to create or process metadata based on the DCAP. Note in particular the following:

- Encoding Schemes and Element Refinements are listed using repeated fields in the Term Usage of the Element to which they refer. In addition, each Encoding Scheme and Element Refinement is also described in its own Term Usage, which allows information about each of them, such as Definition and Constraints, to be recorded in the DCAP as well.

- The Term Usage marks information coming from outside sources: the "Source Definition" copies the definition of Date from DCMI documentation, while the "Local Comment" supplies usage information local to this DCAP.

Annex A: Metadata describing a DCAP

A DCAP should itself be described with Dublin Core metadata, either in a header or in a separate metadata record. At a minimum, this description should include:

Title	A name for the Application Profile.
Contributor	A creator or maintainer of the Profile.
Date	The date of last modification.
Identifier	An unambiguous reference to the Profile. Best practice is to provide a URL by which a copy of the document or schema can be retrieved over the Web.
Description	A concise description of the Profile. As appropriate, the description should elaborate on the context and purposes in which the DCAP is intended to be used; the organizations or individuals involved in its development; any arrangements, policies, or intentions regarding the future development and maintenance of the DCAP; or technical characteristics of the instance metadata or database described.

Annex B: Options for machine-interpretable DCAPs

DCAPs can be expressed in machine-interpretable schema languages, and such machine-interpretable schemas can be manipulated by software applications. This CWA does not give detailed recommendations on how such schemas should be structured, as a number of issues are still open for debate. The scope of this CWA is limited to recommending how application profiles can be expressed as text documents. Future options for machine-interpretable DCAPs are outlined below.

Currently, two schema languages specified by W3C might be considered: XML Schema [XML-SCHEMA] and RDF Schema [RDF-SCHEMA]. The choice of schema language will be influenced by the functionality that the schema is intended to support – for example, whether it is required as a predictable format for data exchange or intended to support inferences about existing metadata. Such different objectives imply different choices between the two schema languages. There has been some discussion on ways to combine XML Schema and RDF Schema to more fully express characteristics of application profiles [HUNTER]. More recently there has been an attempt within the W3C to differentiate RDF Schema as a vocabulary description language and XML Schema as a basis for providing structured data exchange.

An XML schema provides a structured expression that supports validation of instance metadata. In effect, an XML schema provides a document "template" which acts as an exchange format for metadata instances. An XML Schema serves the same function as an XML DTD with additional capability for extensibility and namespace handling.

An RDF schema expresses relationships between terms, providing a data model for expressing the semantics of terms – their properties, classes, and definitions. The underlying RDF data model combined with the use of unique identifiers allows software to infer relationships between terms and perform data aggregation.

RDF Schemas are effective for expressing the semantics of application profiles, whilst XML Schemas are more effective for expressing cardinality, data-typing, and constraints. Possible approaches to the expression of application profiles in RDF have been explored within projects such as SCHEMAS [BAKER] and MEG [MEG-REGISTRY].

Bibliography

[BAKER] Thomas Baker, Makx Dekkers, Rachel Heery, Manjula Patel, Gauri Salokhe, What terms does your metadata use? Application profiles as machine-understandable narratives. Journal of Digital Information 2:2 (November 2001), http://jodi.ecs.soton.ac.uk/Articles/v02/i02/Baker.

[CORES-RESOLUTION] Thomas Baker, Makx Dekkers, Identifying Metadata Elements with URIs: the CORES Resolution. D-Lib Magazine (July 2003), http://www.dlib.org/dlib/july03/baker/07baker.html.

[DC-LIBRARY] Library Application Profile, http://dublincore.org/documents/2002/09/24/library-application-profile/.

[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/.

[DCMI-PRINCIPLES] DCMI Grammatical Principles, http://dublincore.org/usage/documents/principles/.

[DCMI-SCHEMAS] DCMI Schemas, http://dublincore.org/schemas/.

[DCMI-TERMS] DCMI Metadata Terms, http://dublincore.org/documents/dcmi-terms/.

[EGMS] Office of the e-Envoy – Cabinet Office, UK e-Government Metadata Standard Application Profile Version 1, http://purl.oclc.org/NET/eGMSAPv1.

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

[HUNTER] Jane Hunter, Carl Lagoze, Combining RDF and XML Schemas to enhance interoperability between metadata application profiles. WWW10, May 1-5, 2001, Hong Kong, http://www10.org/cdrom/papers/572/index.html.

[MEG-REGISTRY] Rachel Heery, Pete Johnston, Dave Beckett, Damian Steer, The MEG Registry and SCART: Complementary Tools for Creation, Discovery and Re-use of Metadata Schemas. In: Proceedings of the International Conference on Dublin Core and Metadata for e-Communities, 2002. Florence: Firenze University Press, 2002, pp. 125-132, http://www.bncf.net/dc2002/program/ft/paper14.pdf.

[RDF-SCHEMA] Brickley, Dan and Guha, R.V, editors. RDF Vocabulary Description Language 1.0: RDF Schema W3C Working Draft 23 January 2003, http://www.w3.org/TR/rdf-schema/.

[RDN] Powell, Andy, RDN OAI Application Profile, [URL to be assigned].

[RENARDUS] Metadata Working Group SUB Goettingen, Renardus Application Profile, http://renardus.sub.uni-goettingen.de/renap/renap.html.

[URI] T. Berners-Lee, R. Fielding, L. Masinter, Uniform Resource Identifiers (URI): Generic Syntax, August 1998, http://www.ietf.org/rfc/rfc2396.txt.

[WEBARCH] Ian Jacobs, ed., Architecture of the World Wide Web, http://www.w3.org/TR/webarch/.

[XML-SCHEMA] Thompson, Henry S. et al., editors. XML Schema Part 1: Structure. W3C Recommendation 2 May 2001, http://www.w3.org/TR/xmlschema-1/.

Proposed text of the agreement between DCMI and the CEN/ISSS Workshop on MMI-DC

1. CEN will make available the CWAs that are produced by its Workshop MMI-DC for download on the web, under the usual CEN conditions for download (for personal use only, etc). DCMI may include a link to the CWA at the CEN Website

2. DCMI may, but is not obliged to, assign an official DCMI status to documents produced and agreed by CEN Workshops, specifically DCMI Recommendation or DCMI Recommended Resource. DCMI will clearly indicate that such documents have been produced in the context of CEN. Availability of these documents will be through linking to the CWAs concerned on the CEN web-site.

3. CEN grants a royalty-free copyright licence to DCMI to modify CEN CWAs, with full acknowledgement of the original source, and to put these modified ones on the DCMI Web site

4. DCMI commits to provide feedback to CEN on the CWA so that the CWA can be updated through CEN procedures. CEN will take the necessary steps, through its Workshop procedure, to update its CWAs, when requested by DCMI.

5. During development of a CWA, CEN grants DCMI the right to post working drafts on the DCMI Web site and invite public comment and contributions, with a clear indication that this work takes place in the context of CEN. DCMI will channel comments from the DCMI community to the CEN Workshop.

6. In such cases as CEN were no longer able to maintain a Dublin-Core-related CWA, DCMI may, but is not obliged to, take over the maintenance and further development of such document under the normal rules and procedures governing work within DCMI, acknowledging that the original version of the document was produced in the context of CEN and referring to the latest official CWA version.

7. DCMI may, in consultation with CEN, propose a CWA - prepared in the context of CEN and endorsed as a DCMI Recommendation - for consideration by global standards organizations such as ISO and IETF, for approval as an international standard, Internet Standard or RFC.

8. This agreement will be reviewed between CEN and DCMI at the time when WS-MMI-DC would close down but in any case not later than March 2005.

Topic: PBCore - Public Broadcasting Metadata specification
Modified: 2004-02-25 15:44, Wednesday
Maintainer: Tom Baker
Latest version: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/profiles-pbcore/
See also: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/

Shepherd: Diane Hillmann

SUMMARY (Tom)

In February, the (US) Corporation for Public Broadcasting put out a Public Broadcasting Metadata specification for review (see below). The specification is available at http://www.utah.edu/cpbmetadata/PBCore/ -- however, only as a series of Web pages, which makes it problematic to include in the packet. Something will be included in the supplementary packet to be generated on March 11.

Date: Mon, 16 Feb 2004 13:03:51 +0100
From: Thomas Baker <thomas.baker@bi.fhg.de>
To: DCMI Usage Board <dc-usage@jiscmail.ac.uk>
Subject: BATH Agenda: PBCore Metadata Dictionary?

The Corporation for Public Broadcasting in the US wants to use and promote the use of a "PBCore Metadata Dictionary" based on Dublin Core [1]. They are currently (in February) in a comment period, so this is a good opportunity to provide feedback and perhaps nudge them in some helpful directions.

Several issues jump off the page:

1) They use the "dotty" naming style, implying they have HTML in mind.

2) They want to refine the CCP elements but propose to do this with Creator.Role, Contributor.Role, and Publisher.Role.

3) They propose to refine Relation by creating a tag Relation.Type, which would hold a controlled vocabulary of types such as Has Version -- as opposed to Relation.Identifier, which is to hold the identifier.

I would suggest we put this on the agenda with the goal of providing a someone expanded version of the above list, with pointers to better practice. For example, the explanation that Diane and Rebecca are working on about the use of MARC Relator terms could form part of that feedback.

The task of the shepherd will be to formulate the UB comments as a two-page email to the drafters of this proposal.

From: Marcia Brooks <marcia_brooks@wgbh.org>
Sent: Monday, February 02, 2004 6:18 PM
Subject: Request for Input: Public Broadcasting Metadata Initiative
---

Dear Colleague,

For the past several years, a team of public radio and television producers and managers, archivists and information scientists, called together by the Corporation for Public Broadcasting, have been meeting to develop a single, shared protocol for identifying and describing our rich media assets.

Administered by WGBH/Boston, the Public Broadcasting Metadata Initiative <http://www.utah.edu/cpbmetadata/>; is a cross-organizational, multi-disciplined effort to establish a standard way to describe all public broadcasting content (radio and television), so that it can be more easily exchanged between colleagues, software systems, institutions, community partners, individual citizens, etc. It is our belief that in a rapidly evolving and deeply challenging media environment, a well-formed Metadata Dictionary directly addresses our core mission of serving the people of the United States. By working hard to sensibly describe public broadcasting content with standardized, accurate descriptions, we can facilitate easy access and use by teachers, scholars, lifelong learners, engaged citizens and community partners.

We are now at a point where we need to validate our work by gaining the insights of key representatives from both within and outside of the public broadcasting community, and we would like to solicit your assistance in this validation process.

We have created an online survey that is linked to the Public Broadcasting Metadata Dictionary (PBCore). Each PBCore metadata element (which is based on the internationally-accepted Dublin Core format) is described along with its attributes and controlled vocabularies. Your feedback as to the clarity and usefulness of each element will be gathered through this online process - which can be broken up into multiple sessions should you desire.

We hope you will be willing to help us in this important process. Should you wish to substitute a person in your organization or business who is more familiar with content exchange and/or metadata issues, please feel free to indicate their name, phone and e-mail address in your reply to this e-mail. Please reply by February 9 to our project consultant, Mr. Steven Vedro of Madison, Wisconsin, at <srvconsult@charter.net>.

Upon your confirmation of participation, Steven will contact you (or your delegate) with additional details, and you will be welcome to participate at any time during the survey process, which will be conducted from February 16 through February 25.

Thank you in advance for your consideration and assistance.

Marcia Brooks, Project Director, Public Broadcasting Metadata Initiative WGBH/Boston

Topic: Review of application profiles
Modified: 2004-02-25 15:44, Wednesday
Maintainer: Tom Baker
Latest version: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/profiles/
See also: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/

Shepherd: Tom Baker

SUMMARY

In principle, the Usage Board can review application profiles. There are two sets of issues here:

1) Documentation of principles and process

The issue here is to clarify which documents are being followed and whether the existing guidelines are sufficiently clear to allow us to get started with actually reviewing Application Profiles (e.g., at the Shanghai meeting).

-- http://dublincore.org/usage/documents/process/#six
Section 6 of the Process document outlines the procedures by which we can assign the status of "conforming" to application profiles.

-- http://dublincore.org/usage/documents/profiles
"DCMI Usage Board Review of Application Profiles"
This two-page document establishes that the Usage Board can review Application Profiles.

-- http://dublincore.org/usage/meetings/2004/03/cwa14855-20040210.pdf
These CEN "Guidelines for Dublin Core Application Profiles", discussed in another topic, could potentially function as format guidelines for Application Profiles submitted to the Usage Board for review.

2) Actual Application Profiles in the pipeline

The issue here is to compile a list of application profiles that are potentially candidates for review. There are links in my notes but it would take awhile to pull them together; what follows is just a start (please submit additional pointers):

-- http://dublincore.org/documents/2002/09/24/library-application-profile/
Not sure if this is the latest version and whether the Working Group is still in principle interested in a Usage Board review.

-- e-GMS v3 DRAFT for public consultation
http://www.govtalk.gov.uk/schemasstandards/metadata_document.asp?docnum=840

-- Public Broadcasting Metadata specification

http://www.utah.edu/cpbmetadata/PBCore/

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/.

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

Title: # DCMI Usage Board (UB) Administrative Processes (Part 6 only!)

Creator: Diane I. Hillmann, dih1@cornell.edu
Stuart A. Sutton, sasutton@u.washington.edu

Date Issued: **2003-02-07**

Identifier: **http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/CACHE/Process6.html**

Replaces:

Is Replaced By: Not Applicable

Latest version: http://www.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.

---

6. Proposals for Registration of Application Profiles [top]

- **6.1.** Sources of proposals
  - **6.1.1.** DCMI working groups
    - **6.1.1.1.** Existing working groups
    - **6.1.1.2.** Working groups established for the purpose of developing proposals
  - **6.1.2.** Metadata implementers
  - **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 previously reviewed application profiles as models for their layout, appearance, and content. Aside from the required term and element set identifiers, there are no particular constraints on the types of documentation -- local definitions, comments, constraints, or technical notes -- that may be associated with a term.

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'.

6.4.2. 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.

Topic: AskDCMI
Modified: 2004-02-25 15:44, Wednesday
Maintainer: Tom Baker
Latest version: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/askdcmi/
See also: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/

Shepherd: Diane Hillmann

Background

After the AskDCMI service [1] had been operational for a year or so, a number of related issues were discussed in Ithaca in June 2003 [2,3].

In February 2004, Tom raised a number of questions regarding the maintenance of AskDCMI content and of the Usage Board role in its regard [4]. In particular, he suggested that there could be a stronger relationship between the AskDCMI service and DCMI's FAQ [5].

[1] http://askdcmi.askvrd.org/
[2] http://dublincore.org/usage/meetings/2003/06/AskDCMI.html
[3] http://askdcmi.askvrd.org/services/askdcmi/expert_tips.asp
[4] http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/CACHE/AskDCMI.txt
[5] http://dublincore.org/resources/faq/

Date: Fri, 20 Feb 2004 16:42:45 +0100
From: Thomas Baker <thomas.baker@BI.FHG.DE>
Subject: An analysis of questions on AskDCMI
To: DC-USAGE@JISCMAIL.AC.UK

Note: This file,
http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/CACHE/AskDCMI-questions.txt,
to be archived as
http://dublincore.org/usage/meetings/2004/03/AskDCMI-questions.txt.

Dear all,

I took two hours to go through all of the categories on AskDCMI
to see what has been happening, and here are some quick notes
on the 136 questions asked there to date which I submit as
a contribution to the discussion in Bath (see below; this
analysis will also be archived in the directory of materials
for the Bath meeting and will appear at the next build).

The official DCMI FAQ, on the other hand, currently has 31
questions, and some of these are very obviously out of date
(e.g., "What is the Warwick Framework?" is not a frequently
asked question, though one person on AskDCMI did in fact
ask it).

My gut feeling is that we could come up with perhaps 15
broad questions which cover everything in the scope both
of the current FAQ and of the AskDCMI corpus of questions.
I have some ideas how this could be done.

I was also noticing that each of the questions in the current
DCMI FAQ has, in effect, a citable URL, such as:

    http://dublincore.org/resources/faq/#whatismetadata
    http://dublincore.org/resources/faq/#dcmetadata

I'm thinking that if we had a fifteen-question FAQ in which
each question were citable in the manner above, the AskDCMI
experts could answer alot of the questions simply by pointing
people to the URL anchor of the corresponding question in
the FAQ.

I am also thinking that such a FAQ could become the focus
of ongoing Usage Board attention -- an eight-page document
that every Usage Board member is expected to re-read before
every meeting and which then gets explicitly re-approved by
the UB at every meeting.

Tom

Home - About DCMI
    Which countries use it?
    Does DC replace AACR2?
    What is CEN?
    Do search engines index it?

Home - About DCMI - Administration
    What is the IP of DCMI documentation?

Home - About DCMI - Conferences
    None.

Home - About DCMI - Boards/Committees/Working Groups
    What ever happened to Admin Core?

Home - About DCMI - Documentation
    What is the URL for the NISO standard?
    (Garbled question.)
    How do I use DC for photograph collections?
    Does it cost money to use DC?
    Can you recommend an article about people's names?
    What are refinements?

Home - Implementation
    To embed or not?
    How can I use DC for content management and search in the newspaper industry?
    How do I implement DC in MS-Access?
    Do DCMI documents tell you how to generate tags on the fly?
    Is there a classification scheme for still photos?
    Is there a tool to convert USMARC to DC?
    Can I use DCMI's "document status" element?
    How can I put centuries and ranges into dc:date?
    Is there a (DCMI) standard for keywords?
    Where can I find translations of Dublin Core?
    How can I describe geographic information?
    How can dc:source be used to reference old books?
    Was Creator.Personalname.Address approved somewhere?
    Could we register the European Music Navigator vocabulary?
    Is there a list of documentation types for software?
    How do we describe conference proceedings with DC?
    Four questions on search engines, tags, and databases.
    How can I use DC to describe a collection of photographs?
    What ever happened to DC.Creator.CorporateAddress?

Home - Implementation - Metadata in general
    What is the difference between a thesaurus and a classification?
    What is the difference between a thesaurus and a taxonomy?
    How is metadata used in bibliographic networks?
    Where can I take a course on DC?
    Must we pay DCMI if we provide a drop-down menu for dc:subject?
    "WHAT IS METADATA IN VIDEO?" by JOACHIN
    Why do search engines have trouble with my XML tags?
    Does Google index DC?
    What does CODEN stand for?
    What is the history of metadata?
    Which search engines use DC in HTML headers?
    Does DC have to go on every Web page?
    Are there any enterprise metadata warehouse applications?
    "Metadata framework implementation" - How do I analyse the structure of a resource?
    Where can I find the education extensions for DC?
    In "isVersionOf", why are the words run together?
    Whatever became of the Warwick Framework?

Home - Implementation - Maintenance

Home - Implementation - Other issues
    Is Audience the 16th element?
    How can I use DC to search for photographs?
    What is the relationship between the CORES Project and DCMI?
    What is the difference between DC and SCORM?
    How can I use DC for spatial metadata?

    Do DCMI documents tell you how to generate tags on the fly?
    Should I translate the tag "creator" into Vietnamese?
    Where can I take classes on DC, XML, and RDF?
    Where can I find a standard approach to versioning documents?
    What is the relationship between DC and FRBR?

Home - Implementation - Storage
    How can I store metadata separately and link?
    How could DC be implemented in RDBMS?

Home - Terms
    Could DC help us describe art-history slides?
    Is Audience the 16th element?
    What is the relationship between DCMI and the IEEE/LOM community?
    Is there a difference between a sub-element and a refinement?
    What is the semantics of Title?
    What is the difference between a Term and an Element?
    Can I add a field to an RSS feed?
    What are sub-elements?

Home - Terms - Application Profiles
    Does the creator of an AP decide what should be mandatory?

Home - Terms - Encoding Schemes
    Disappearing angle brackets: http://askdcmi.askvrd.org/index.asp?id=5239
    How do I register an encoding scheme?
    I'm confused about the example in the DCSV recommendation.

Home - Terms - Encoding Schemes - Syntax Encoding Schemes
    Very specific question about DCMI Point

Home - Terms - Encoding Schemes - Vocabulary Encoding Schemes
    Will DCMI register additional vocabulary schemas?
    Disappearing angle brackets: http://askdcmi.askvrd.org/index.asp?id=5239
    How can I register a vocabulary?
    Is there a standard markup language for controlled vocabularies?
    Must we pay DCMI if we provide a drop-down menu for dc:subject?
    Which of the DCMI types would I use for cataloging a Website?
    I am looking for subject taxonomies for eGovernment.

Home - Terms - Encoding Schemes - Vocabulary Terms
    Where do I put video track information?
    What are sub-elements?

Home - Terms - Elements
    Can I use dc:identifier for ISSNs?
    Distinction creator/publisher.
    Is Audience the 16th element?
    Where do I put the name of the person submitting the material?
    Four different questions about dc:relation.
    Someone told me there are 16 elements.
    Where do I put the location of the creator?
    Where do I put the location of the publisher? (different person)
    How does DC relate to RSS?
    A follow-up email on the four questions about dc:relation.
    What is the semantics of Title?
    How do I use DC with SCORM and DMSO/HLA?
    Are all elements repeatable?
    Which elements are mandatory?
    How can I add an element?
    What equivalencies exist between DC, IEEE/LOM, and SCORM?
    Can I repeat dc:identifier?

    What is the difference between "term" and "element"?
    Should I use "isPartOf" or "isReferencedBy"?
    Is Dublin Core extensible?
    Please give me examples of Contributor.
    How can I say something is a Press Release?
    Should I repeat dc:publisher and distinguish between the two types?
    How can I implement name authority?
    How do I describe Sponsors and Partners of a project in RDF?
    How do I indicate the source of a classification code?

Home - Terms - Refinements
    What is the difference between sub-elements and refinements?
    Can I create my own refinements?
    What are sub-elements? (someone else)
    Are multi-element tags like "Date.Created.Digital" interoperable?
    How can I extend Creator to put in birthdates?
    How can I extend Creator to distinguish first name from family name?
    How can I use XSLT to create multi-element DCSVs?
    Can I put number of pages into Format?

Home - Syntaxes
    Where can I find the xsd files that work together?
    Which is better, HTML or XML?
    Which is better, XML or HTML?
    Which is better, XML or RDF?
    How do I use DC for geographic information?

Home - Syntaxes - HTML
    What is the semantics of Title?
    Are keywords case-sensitive?
    Should it be "creator" or "Creator"?

Home - Syntaxes - RDF
    How do I add a field to an RSS feed?
    What is the correct RDF syntax for "isPartOf"?
    Which of these two syntaxes is correct?
    Can the value of dc:subject be a URI representing a person?

Home - Syntaxes - Other
    What are the intended cardinalities?
    Disappearing angle brackets: http://askdcmi.askvrd.org/index.asp?id=5239
    Can one specify a minimum length of 1?
    Where can I find the xsd files that work together?
    Can DC in XML have qualifiers?
    Should we translate the tags into Marathi?

Home - Syntaxes - XML
    Are the XML schemas "official" and stabile?
    How can I dumb down Dc.creator.contact?

Date: Tue, 10 Feb 2004 10:55:04 +0100
From: Thomas Baker <thomas.baker@bi.fhg.de>
Subject: AskDCMI on the Bath agenda?

I see the questions building up on AskDCMI again and am feeling
uncomfortable about the model here, so I'm thinking we should
perhaps give this issue some time in Bath.

Some of what is making me uncomfortable:

-- When questions come into my mailbox but not answers,
   I cannot know which ones have been answered without going
   to the Web, nor can I accumulate my own archive of answers,
   meaning I have to search the database on the Web to know
   whether a similar question has been answered.

-- Not being able to get an overview of what has been written,
   I am uneasy with the notion of a growing set of overlapping
   answers to overlapping questions, and that some of these
   answers may in fact be controversial, or out-of-date,
   or at any rate need to be maintained.

-- It is not clear to me whether the goal is to accumulate a
   body of best practice and use that corpus to refine and
   improve our collective sense of best practice over time.

   If the goal is not to accumulate knowledge, then the AskDCMI
   work feels like a relentless servicing of user requests.

   If the goal is to learn from the process, then it is not
   clear to me how we can capture that learning. For example,
   if the answers were being cited and summarized in a
   single, growing FAQ file that we could periodically
   download and keep on hand, then one could presumably
   answer alot of the new questions simply by pointing to
   a specific section of the FAQ. This would also have
   the welcome side effect of improving the quality of
   http://dublincore.org/resources/faq/, which does indeed
   need to be revised.

-- If the answers are not being captured or summarized in such
   a central document, it is not clear to me what longer- term
   responsibility is being implicitly assumed (and by whom)
   for reviewing and maintaining the many separate answers.
   In principle, does an answer have an expiration date?

-- It is not clear to me how AskDCMI relates to DCMI's
   open mailing lists, where asking the same question might
   lead to a discussion instead of a straightforward (but
   possibly simplistic or misleading) answer.

I pose this concerns in a constructive spirit because I would
very much like to see this service work. On the other hand,
AskDCMI is not a service of the Usage Board, so I am uncertain
how far this discussion should go. In Bath, we could perhaps
consider the following:

-- A printout (if such can be generated) of the entire corpus
   of answers and rough analysis of questions by type.

-- A general discussion of current plans and future
   developments in light of the comments above.

Topic: Proposals for dc:rights-related terms
Modified: 2004-02-25 15:44, Wednesday
Maintainer: Tom Baker
Latest version: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/terms-rights/
See also: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/

Shepherd: Andrew Wilson

SUMMARY (Tom)

A proposal was submitted for two rights-related terms

http://www.ukoln.ac.uk/metadata/dcmi/dc-rights/

There has been considerable discussion on the list, which Andrew will summarize at the meeting.

One parallel development we should be aware of is an effort to create an Rights schema for use in OAI:

http://www.openarchives.org/documents/OAIRightsWhitePaper.html

This short white paper, which refers to dc:rights, will be included in the supplementary packet, though it may not be relevant enough to the approval of the current proposals to discuss in Bath.

Proposals for DC Rights-related Terms

Stuart Weibel
OCLC Office of Research

Andy Powell
UKOLN, University of Bath

Eric Miller
World Wide Web Consortium

February 11, 2004

Introduction and background Information

The DC Rights element was added to the Dublin Core following DC-3, in recognition of the importance of terms and conditions of use for discovered resources. To date, it has been little utilized, owing to the lack of standard practice concerning rights declarations on the Internet.

The recent emergence of the Creative Commons as a clearinghouse for rights declarations affords an opportunity to improve this situation, particularly for resources that have been developed with the intention of cost-free distribution, but whose creators wish to formally declare various rights.

Creative Commons has defined several licenses from which content rights holders may choose. Each of these licenses is unambiguously identified with a URI that is managed as unchanging and persistent by Creative Commons. Thus, this namespace comprises a controlled, enumerated vocabulary of license declarations, open for use by any party for whom their terms and conditions are judged useful. Providing a clear method for embedding of CC license information within the Dublin Core will reinforce the impact of both protocols. Both CC proponents and DC adopters will benefit by having a clear approach to formal rights declaration in a widely adopted metadata framework on the Internet. Further, the model for such declarations has been defined so that it is broadly useful for declaring licenses from other sources as well, providing a general-purpose mechanism for intellectual property declarations.

This proposal outlines functional requirements, discusses alternative representations, and proposes a standard of practice.

Statement of requirements

DCMI metadata currently supports the following functional requirements, through the use of the DC Rights property:

- provide a simple human-readable statement of who holds rights over a resource
- support simple (i.e. potentially ambiguous) searches of the form "find all resources where an entity that is named using a simple string is likely to be a rights holder"

By using appropriate conventions for the value string of dc:rights (e.g. "(c) Copyright University of Bath, 2002") it is possible to meet narrower functional requirements, such as "provide a simple human-readable statement of who holds copyright over a resource" and "support simple (i.e. potentially ambiguous) searches of the form 'find all resources where an entity that is named using a simple string is likely to be the copyright holder'". However, these narrower requirements can only be met using searches based on simple string comparisons (e.g. search for '"copyright" AND "University of Bath"'), which is not a very robust approach.

The two proposals for new DCMI properties below are intended to support the following additional functional requirements:

- Support unambiguous searches of the form "find all resources where an entity named using a simple string is a rights holder"
- Support unambiguous searches of the form "find all resources where an entity that is identified using a URI is a rights holder"
- Provide an unambiguous statement of the license under which the resource is made available, based on the URI associated with that license
- Support unambiguous searches of the form "find all resources which are made available under the terms of the license with the following URI"

Furthermore, there may be additional requirements related to specific kinds of rights (copyright, patent rights, database rights, etc.). For example:

- support unambiguous searches of the form "find all resources where an entity named using a simple string is the copyright holder"
- support unambiguous searches of the form "find all resources where an entity that is identified using a URI is the copyright holder"

These more specific requirements are not met by the two proposals below. However, they could be met through the proposal of narrower sub-properties of dcterms:rightsHolder, for example dcterms:copyrightHolder.

Proposed Element Refinements

There are two primary requirements for rights declaration addressed in this proposal:

1. identifying the license associated with a particular resource, and
2. identifying the person or organization holding the rights.

This proposal recommends the adoption of the following terms to meet these requirements:

license

rightsHolder

The specifics of each proposed term are elucidated in the Proposal Requirements Tables below.

Impact on the "dumb-down" principle

Anticipated values for the proposed elements remain useful when stripped of their qualifiers, however, as with all such "dumbed-down" metadata values, the results will not be interpretable with the same degree of confidence. This is particularly important in the treatment of rights metadata because of the potential legal implications. A "dumbed-down" rights statement may be incomplete, but should not mislead.

Implications for other license schemes

This proposal was motivated by the requirements for supporting declaration of Creative Commons licenses, but can be similarly applied to any set of licenses without future coordination with DCMI or changes to DCMI metadata terms. Any party desiring to specify and maintain licenses can take advantage of this infrastructure.

DCMI Decision Tree Supporting Information:

Clarity of semantics and expression

Terms proposed in this proposal are defined with a degree of specificity and constraint equal to, or greater than, other DCMI terms.

It is recommended that the value of *license* is denoted by a URI identifying an IPR license.

The *rightsHolder* term is defined in much the same manner as any of the agent elements of DC (creator, contributor, and publisher), with an additional constraint that the identified entity must be able to be associated with intellectual property rights (thereby ruling out instruments as valid values for this term).

Practicality

Use of Creative Commons licenses represents a formal opportunity for right holders to declare terms and conditions outside the framework of commercial Digital Rights Management (DRM) systems, and as such has the potential to have an important impact on the development of best practice in communities managing digital assets. In addition, the proposal is written such that licenses other than Creative Commons licenses can be declared without retrofitting these terms, and without requiring other organizations or initiatives to adopt namespaces from competitive initiatives.

It is expected that much commercially managed content will increasingly be packaged within DRM systems. The current proposal is based on the declaration of digital rights, rather than management, and is more suitable for organizations and individuals that want to make their content widely available, often without compensation, but without foregoing specified rights.

Specifying license URIs is straightforward and unambiguous, taking advantage of the basic Web infrastructure, and relying on the commitment of organizations to support such infrastructure over time.

Appropriate Placement within the DC Architecture

The proposed *license* term refines the Rights element, affording a degree of specificity and constraint that make it broadly useful for rights declaration applications. The proposed *rightsHolder* term is not a refinement of any existing DCMI terms. Declaring the new terms within the DCMI namespace will simplify their interpretation by metadata applications and provide sufficient generality to support any group that chooses to specify licenses. No coordination with DCMI would be necessary to expand the use of the terms by other organizations in the future, and DC will help to establish a consistent and useful deployment framework for the community at large.

An alternative to the proposed approach is to declare a Creative Commons encoding scheme and simply register it as such, leaving the semantics, structure, and syntax to the Creative Commons community. This would work, but is undesirable in that there would be little incentive for competing licensing schemes to use CC's encoding, possibly leading to divergence in the structure, encoding, and semantics of multiple rights declaration schemes.

Creative Commons is an organization that is focused on licenses, not on the structure and encoding of the metadata about those licenses. DCMI, in turn, has no expertise in licenses, but can facilitate rights declaration by promoting a consistent and flexible means for encoding declarations.

Existing implementations

Examples of organizations using CC licenses can be found at

http://creativecommons.org/learn/collaborators

Those participating include a variety of organizations that come from traditional DCMI constituent communities.

Decision Tree Table

1. While the specification of a Creative Commons encoding scheme would satisfy the immediate need for declaration of a Creative Commons license, such an approach suffers from two significant disadvantages outlined earlier: separation of organizational function: (a) Creative Commons is concerned with licenses, DCMI is concerned with maintaining metadata terms, and (b) such an approach will do nothing to promote consistent means for rights declaration in the community at large, and hence will not come into wide use by harvesters.
2. The use of an application profile for specification of rights declaration suffers from precisely the same problems as encoding schemes.
3. All valid values of the proposed *license* term will also be valid values of the existing DC Rights term - therefore this is proposed as an element refinement of DC Rights.
4. All valid values of the proposed *rightsHolder* term will not be valid values for any existing DCMI term - therefore this is proposed as a new element.

Proposal 1 - rightsHolder

Name
: rightsHolder

Label
: Rights Holder

Definition
: A person or organization owning or managing rights over the resource.

Comment
: Recommended best practice is to use the URI or name of the Rights Holder to indicate the entity.

Examples
: Stuart Weibel
: http://w3.org/
: University of Bath

Type of term
: DC term

Term qualified
: none

Why needed
: To support the identification of rights holders, either default copyright holders or a person or organization that is legally empowered to assign and transfer licenses to the relevant resource.

Working Group support
: This proposal has been initiated within the DCMI Directorate in response to an identified opportunity for the improvement of intellectual property rights declaration for Internet accessible resources. The proposal has been developed in consultation among Stuart Weibel of the DCMI Directorate, Eric Miller of the W3C, technical advisor to the Creative Commons, and DCMI Advisory Board member, and Andy Powell, Chair of the DC-Architecture working group.

Proposed status
: The term is proposed as Recommended.

Related DCMI terms
: None

Related non-DCMI terms
: None Identified

Impact on applications
: Potentially improved consistency for digital rights declarations

About the proposers
: Stuart Weibel OCLC Office of Research
: Eric Miller W3C Semantic Web Activity Lead
: Andy Powell UKOLN, DCMI Usage Board

Proposal 2 - license

Name
: license

Label
: License

Definition
: A legal document giving official permission to do something with the resource.

Comment
: Recommended best practice is to identifiy the license using a URI. Examples of such licenses can be found at: http://creativecommons.org/licenses

Examples
: http://creativecommons.org/licenses/by/1.0/

Type of term
: Element refinement

Term qualified
: dc:rights

Why needed
: To support the identification of formal licenses associated with resources.

Working Group support
: This proposal has been initiated within the DCMI Directorate in response to an identified opportunity for the improvement of intellectual property rights declaration for Internet accessible resources. The proposal has been developed in consultation among Stuart Weibel of the DCMI

Directorate, Eric Miller of the W3C, technical advisor to the Creative Commons, and DCMI Advisory Board member, and Andy Powell, Chair of the DC-Architecture working group.

Proposed status

The term is proposed as Recommended.

Related DCMI terms

There are no existing or proposed qualifiers for the DC Rights element; the present proposal proposes the following term: license.

Related non-DCMI terms

None identified

Impact on applications

Potentially improved consistency for digital rights declarations

About the proposers

Stuart Weibel OCLC Office of Research
Eric Miller W3C Semantic Web Activity Lead
Andy Powell UKOLN, DCMI Usage Board

Topic: MARC Relator terms as sub-properties of dc:contributor
Modified: 2004-02-25 15:44, Wednesday
Maintainer: Tom Baker
Identifier: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/terms-relators/
See also: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/

Shepherd: Rebecca Guenther

SUMMARY (Tom)

This topic seems to have three threads:

-- Ongoing discussion of what it really means when MARC Relator terms are declared to be semantically narrower than dc:contributor (see the rough-cut email excerpts below).

-- Guidelines for using Agent Roles in Dublin Core -- a draft prepared by Diane and Rebecca and printed in the main packet: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/CACHE/Agent-Roles-Guidelines2.txt.

-- The mechanics of declaring machine-processably that all (or maybe just most!) Relator terms are subPropertyOf dc:contributor. The 13-page RDF schema, http://www.loc.gov/marc/dc/marcrelcodes.rdf, will be put into the supplementary packet for reference, though the nitty-gritty of the RDF encoding should be largely out of scope for our discussion in Bath.

Date: Wed, 24 Sep 2003 13:25:23 -0400
From: "Rebecca S. Guenther" <rgue@LOC.GOV>
Subject: MARC relators in RDF
To: DC-USAGE@JISCMAIL.AC.UK

I have put up the MARC relator code list in RDF at:
http://www.loc.gov/marc/dc/marcrelcodes.rdf

I read over the comments about equivalent terms and decided to do the following:

1. All relator terms have the following:
<rdfs:subPropertyOf rdf:resource="http://purl.org/dc/terms/contributor" />

Meaning that it is equivalent to dc:contributor.

[** Note: Andy suggested that Rebecca meant to say "subPropertyOf dc:contributor" here. **]

2. marc.relators/cre (i.e. marc:creator) has the following statement:
<owl:equivalentProperty
rdf:resource="http://purl.org/dc/elements/1.1/creator" />

This is what Roland suggested. I felt that this was better than sameAs, since I understood this to be a weaker equivalence. Because marc:creator and dc:creator have different semantics, I thought this was safer.

3. I made no statement about equivalence between marc:publisher and dc:publisher, because the relator terms only are used when the entity functions as a publisher that makes contributions. dc:publisher, as has been pointed out, makes the resource available. This term indicates contributions to the resource in the context of also being a publisher. So marc:publisher just has the statement as above concerning its equivalence to dc:contributor.

We can discuss this further at the UB meeting in Seattle and any other issues about implementing these terms as refinements of dc:contributor.

FYI: the above URL will be changed at some point in the future when we set up a namespace for the relators.

Date: Mon, 7 Jul 2003 11:11:03 -0400
From: "Diane I. Hillmann" <dih1@CORNELL.EDU>
Subject: Re: Comments on Encoding Scheme registration
To: DC-USAGE@JISCMAIL.AC.UK

Yes, I think the important thing to remember is that we decided to include all roles on the LC relator list so that implementers could make the choice of what to use, rather than US having to anticipate their needs. I think when we get to the point of expressing our decision and how people ought to use these terms, the idea that the choices are theirs needs to be made.

Date: Mon, 7 Jul 2003 17:52:44 +0200
From: Roland Schwaenzl <Roland.Schwaenzl@MATHEMATIK.UNI-OSNABRUECK.DE>
Subject: Re: Comments on Encoding Scheme registration
To: DC-USAGE@JISCMAIL.AC.UK

The issue is NOT, whether they CAN be contributors in some cases, but whether they ARE contributors in all possible cases.

marc:publisher as you put it in NOT redundant to a dc:publisher. A dc:publisher is allowed simply to make a resource available as is and is not a priori supposed to contribute to it's content.

> They will add other terms
> from the MARC relator list where necessary to bring out a particular role
> in relation to the resource. If they're using MARC they'll use
> marc:publisher (in the way it's used in MARC, which is as a value of a
> relator element associated with the name). My understanding is that we are
> adding these subProperty statements for machine readability, but I don't
> imagine people will get confused.

Information systems will DRAW AUTOMATIC conclusions from
subProperty relations. It is important to get subProperty
relations right.

Date: Mon, 16 Feb 2004 23:50:36 +0000
From: Andy Powell <a.powell@UKOLN.AC.UK>
Subject: Re: Guidelines for the use of Roles
To: DC-USAGE@JISCMAIL.AC.UK

On Mon, 16 Feb 2004, Diane Hillmann wrote:
> Rebecca and I have a draft to present for comment. It lacks examples,
> which our notes suggest Andy was willing to provide. Because we've gotten
> down to the wire (apologies), we decided to distribute the draft and see if
> we've covered the necessary ground, hoping that Andy will flog around some
> examples to be added later.

There remain issues with the use of MARC relator codes - and these have
recently surfaced on the DC Collection Description WG mailing list.

The CD WG need a property to indicate the owner of the resource (in this
case a collection). The use of marc:own was discussed. marc:own is
defined to be a refinement of dc:contributor. However, in this case, the
owner will not necessarily have made a contribution to the collection.
They may simply have bought the collection for example. Therefore, the WG
will have to find an alternative to marc:own.

So, two things...

1) The definition of marc:own at

 http://www.loc.gov/marc/dc/marcrelcodes.rdf

needs to make it crystal clear that in this case, 'owner' is defined to be
an owner that has made contributions to the content of the resource.
I.e., the defintion of marc:own should be changed to something like

 The person or organization that currently owns and has made a
 contribution to the content of the item or collection.

2) In many cases WGs and other groups will have to find alternatives to
the MARC relator codes because the restrictive definitions (as having made
a contribution) are not intuitive and are too narrow for many uses.

Date: Tue, 17 Feb 2004 11:37:39 -0000
From: Pete Johnston <p.johnston@UKOLN.AC.UK>
Subject: Re: Guidelines for the use of Roles

I was going to say I fully agree with Andy's comments below, especially

the first point about the need to make it explicit in the textual definitions of these properties that they imply that the object of the property is an entity that contributes to the content of the resource.

But looking at the list of properties more closely, it seems to me that there are cases where saying that they are a sub-property of dc:contributor introduces an element of contradiction:

e.g. http://www.loc.gov/marc.relators/dub

Label = "Dubious author"
Comment = "A person or corporate body to which authorship has been dubiously or incorrectly ascribed."

So in making a statement using this sproperty, on the one hand you assert that authorship has been dubiously or incorrectly ascribed, but at the same time you say that they are "An entity responsible for making contributions to the content of the resource" (!)

Yes, I can just about see that you _can_ assert that their claim to _authorship_ is dubious, and also assert that they still made some other contribution to the content, without contradicting yourself.

But in practice this seems to limit the legitimate use of the property to the extent that it is almost useless in any practical sense.

I'm probably re-opening a can of worms, but I'll do it anyway: why is it necessary to declare all of these things as sub-properties of dc:contributor? It seems to me that many of them make perfectly useful properties without asserting any relationship to dc:contributor (and indeed asserting that relationship only serves to limit their usefulness).

Date: Tue, 17 Feb 2004 17:30:30 -0500
From: "Rebecca S. Guenther" <rgue@LOC.GOV>

I guess it's because there was the argument that if we use these roles with a DC element, that DC element being contributor, it has to refine the semantics of the element. So I was told we had to put that statement in for each of the role terms on our list (which didn't originally have anything of the sort). And I followed instructions, although we wouldn't be having these arguments about the use of the terms if they didn't make these assertions. So we have a list of stable, standardized terms for roles, but there seems to be this argument that we can't use them if they don't exactly fit in with the semantics of contributor. Maybe the way out is to consider the word "contributions" as broad and inclusive of indirect types of contributions. Maybe the problem is the choice of words we used for elements in the original DCMES. (I'm sure you'll all love that statement.)

Date: Tue, 17 Feb 2004 17:15:59 -0500
From: "Rebecca S. Guenther" <rgue@LOC.GOV>
Subject: Re: Guidelines for the use of Roles

On Mon, 16 Feb 2004, Andy Powell wrote:

> On Mon, 16 Feb 2004, Diane Hillmann wrote:
>
> > Rebecca and I have a draft to present for comment. It lacks examples,
> > which our notes suggest Andy was willing to provide. Because we've gotten
> > down to the wire (apologies), we decided to distribute the draft and see if
> > we've covered the necessary ground, hoping that Andy will flog around some
> > examples to be added later.
>
> There remain issues with the use of MARC relator codes - and these have
> recently surfaced on the DC Collection Description WG mailing list.
>
> The CD WG need a property to indicate the owner of the resource (in this
> case a collection). The use of marc:own was discussed. marc:own is
> defined to be a refinement of dc:contributor. However, in this case, the
> owner will not necessarily have made a contribution to the collection.
> They may simply have bought the collection for example. Therefore, the WG
> will have to find an alternative to marc:own.

marc:own says that it's a refinement of dc:contributor because we were told this had to be done to be able to use these terms.

All of these terms were initially included in the relators list because an institution thought they played a large enough role in connection with a resource to warrant having access to it by that entity's name. Thus, we concluded that this is a form of contribution. In the case of owner, the relationship to the resource is that the named entity is an owner and this role is important to the given resource. So it is a very broad definition of "contributor". We don't actually call them contributors as such in MARC, which was what this list was defined for, but consider them making a contribution to the resource in an indirect way. So for any situation where "owner" is the role that an entity played in the resource, it would be the type of resource for which ownership is important (probably rare material that might not be identified any other way).

> So, two things...
>
> 1) The definition of marc:own at
>
> http://www.loc.gov/marc/dc/marcrelcodes.rdf
>
> needs to make it crystal clear that in this case, 'owner' is defined to be
> an owner that has made contributions to the content of the resource.
> I.e., the defintion of marc:own should be changed to something like
>

> The person or organization that currently owns and has made a
> contribution to the content of the item or collection.
>
> 2) In many cases WGs and other groups will have to find alternatives to
> the MARC relator codes because the restrictive definitions (as having made
> a contribution) are not intuitive and are too narrow for many uses.

I'm not sure I understand this comment. The MARC definitions are not currently so narrow (as I described above). Adding something like what you suggest to the definition will make them narrower than they were intended. That could affect existing usage and invalidate existing records. So I don't think we will want to narrow these definitions in this way.

Date: Thu, 19 Feb 2004 11:49:20 +0000
From: Andy Powell <a.powell@UKOLN.AC.UK>
Subject: Re: Guidelines for the use of Roles
To: DC-USAGE@JISCMAIL.AC.UK

On Wed, 18 Feb 2004, Pete Johnston wrote:

> Right. So that narrowness is _not_ present in the initial MARC relator
> definitions. I wasn't clear whether it was or not.

Good. I'm glad that we seem to have clarified this at last.

> But making the statement (as the declaration sof the RDF properties do):
>
> marcrel:own rdfs:subPropertyOf dc:contributor .
>
> does effectively generate this narrowing, I think.

Yes, that's right. Declaring some of these relators as being sub-properties of dc:contributor does narrow their intended semantics.

> Well, strictly speaking, what it says is, every time I say:
>
> _x marcrel:own _y . (Resource x has-an-owner Entity y)
>
> I also imply:
>
> _x dc:contributor _y . (Resource x has-an-entity-that-made-a
> contribution-to-the content Entity y)
>
> Which I don't think you really want to say.

Agreed.

> I take your point that you can argue that the entities related to a
> resource by a MARC relator property do "make a contribution" to the
> resource in the very general sense you describe above. But (it seems to
> me) they do not necessarily do so in the specific sense of

> dc:contributor.

Agreed.

> Now we might well argue that's a problem with the definition of
> dc:contributor (as I think you suggest in your subsequent message). But
> working on the basis of the current definition of dc:contributor, it
> seems to me that stating the subproperty relation creates an undesirable
> result in many cases.

Agreed.

> So it seems to me the problem is with the constraint that:
>
> > marc:own says that it's a refinement of dc:contributor
> > because we were told this had to be done to be able to use
> > these terms.

Well, somewhere along the line, this whole process seems to have got very confused... which is all water under the bridge, so probably best forgotten?

The important thing to recognise is that the MARC properties can be used irrespective of whether or not they refine DC properties. And indeed, they become much more useful if they are not tied to DC properties in an inappropriate way.

> It seems to me that it makes perfect sense to describe some of these
> relator terms as refinements of dc:contributor, but for others it does
> not. And that decision has to be taken on a case-by-case basis not on a
> one-rule-for-all. But even for those relator terms that are not
> refinements of dc:contributor, it's still useful for LoC to declare them
> as RDF properties so that applications can use them: they just have
> nothing to do with dc:contributor. That's OK, isn't it?

Yes. So... I think the best course of action is to go thru the current LoC RDF schema declaration for these terms and do the following:

- remove rdfs:subPropertyOf assertions where it is not appropriate, e.g. for marc:own
- add explicit owl:samePropertyAs assertions w.r.t. dc:creator, dc:publisher and dc:contributor (there are one or two of these but I suspect there could be more, e.g. marc:aut == dc:creator)

Then we'll have a much more useful set of new properties that don't make wrong or confusing assertions about their relationship to existing DC properties... and which then can be used by other initiatives like the DC Collection Description WG.

Is there any reason not to change the RDFS in this way? I am happy to have a stab at making the changes to the LoC RDFS, but probably better for someone at LoC to do it.

Date: Thu, 19 Feb 2004 09:45:30 -0500
From: "Rebecca S. Guenther" <rgue@LOC.GOV>
Subject: Re: Guidelines for the use of Roles

So, to clarify, if you have a MARC relator term that is NOT a refinement of Contributor, then you just use it without any relation to a DC term? We don't have this problem with semantics in either MARC or MODS because the element we use it with is called "name", which is much broader than DC creator or contributor. In other words, DC has no element that is a name associated with a resource, but those elements stipulate specific roles related to the resource that are inherent in their semantics. I guess this is really the problem. I haven't been following the collection description work, but if there were a resource description and you wanted to include the owner, you would use DC elements for everything except that and use marc:own (or whatever we decide) for the owner's name?

I think what Andy suggests is appropriate. I could work with Andy to do this. Andy, let's talk about how best to get this accomplished.

Date: Thu, 19 Feb 2004 15:11:33 +0100
From: Roland Schwaenzl <Roland.Schwaenzl@MATHEMATIK.UNI-OSNABRUECK.DE>
Subject: Re: Guidelines for the use of Roles

> Yes, that's right. Declaring some of these relators as being
> sub-properties of dc:contributor does narrow their intended semantics.

If the intended semantics is as aluded to above, the declaration is counter productive and will yield unexpected results in retrieval (narrow minded people would say "wrong results").

I'm kind of surprised to read that the MARC relators have a broader semantics. When we discussed the issue last time concerns i had were dismissed on the basis, that narrower semantics is understood, just not explicitly stated till now.

> > Well, strictly speaking, what it says is, every time I say:
> >
> > _x marcrel:own _y . (Resource x has-an-owner Entity y)
> >
> > I also imply:
> >
> > _x dc:contributor _y . (Resource x has-an-entity-that-made-a
> > contribution-to-the content Entity y)
> >
> > Which I don't think you really want to say.

The information given previously was exactly opposite.

> > Now we might well argue that's a problem with the definition of
> > dc:contributor (as I think you suggest in your subsequent message).

When you want to have a different semantics go ahead, provide a
definition for a new dc term and advertise it.

> Well, somewhere along the line, this whole process seems to have got very
> confused... which is all water under the bridge, so probably best
> forgotten?

Back to A ?

> The important thing to recognise is that the MARC properties can be used
> irrespective of whether or not they refine DC properties. And indeed,
> they become much more useful if they are not tied to DC properties in an
> inappropriate way.

We have to take care of the dc vocabulary. We can say,
we find the asserted relation inappropriate and suggest to delete it.

> > But even for those relator terms that are not
> > refinements of dc:contributor, it's still useful for LoC to declare them
> > as RDF properties so that applications can use them: they just have
> > nothing to do with dc:contributor.

Certainly it is useful to have well coined and structured
vocabularies for use in RDF.

It's our bussiness to further promote DC vocabulary in this area.

> Yes. So... I think the best course of action is to go thru the current
> LoC RDF schema declaration for these terms and do the following:
>
> - remove rdfs:subPropertyOf assertions where it is not appropriate, e.g.
>   for marc:own
> - add explicit owl:samePropertyAs assertions w.r.t. dc:creator,
>   dc:publisher and dc:contributor (there are one or two of these but I
>   suspect there could be more, e.g. marc:aut == dc:creator)

Not to use owl "sameAs" we decided previously.

Date: Fri, 20 Feb 2004 10:21:13 -0000
From: Pete Johnston <p.johnston@UKOLN.AC.UK>
Subject: Re: Guidelines for the use of Roles
To: DC-USAGE@JISCMAIL.AC.UK

The relevant paragraph is here

http://www.w3.org/TR/2004/REC-owl-ref-20040210/#equivalentProperty-def

NOTE: Property equivalence is not the same as property equality.
Equivalent properties have the same "values" (i.e., the same property
extension), but may have different intensional meaning (i.e., denote

different concepts). Property equality should be expressed with the owl:sameAs construct. As this requires that properties are treated as individuals, such axioms are only allowed in OWL Full.

But I'm not at all sure I understand the distinction being made.

Roland, could you maybe explain the distinction with some examples of what it would mean if we used owl:equivalentProperty and what it would mean if we used owl:sameAs please?

Date: Mon, 16 Feb 2004 23:36:31 +0000
From: Andy Powell <a.powell@UKOLN.AC.UK>
Subject: Re: Guidelines for the use of Roles
To: DC-USAGE@JISCMAIL.AC.UK

On Mon, 16 Feb 2004, Diane Hillmann wrote:
> Rebecca and I have a draft to present for comment. It lacks examples,
> which our notes suggest Andy was willing to provide. Because we've gotten
> down to the wire (apologies), we decided to distribute the draft and see if
> we've covered the necessary ground, hoping that Andy will flog around some
> examples to be added later.

XHTML

```
<link rel="schema.DC" href="http://purl.org/dc/elements/1.1/" />
<link rel="schema.MARC" href="http://www.loc.gov/marc.relators/" />

<meta name="DC.title" content="Reichstag" />
<meta name="MARC.arc" content="Sir Norman Foster" />
```

XML

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

<metadata
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:marc="http://www.loc.gov/marc.relators/">
  <dc:title>Reichstag</dc:title>
  <marc:arc>Sir Norman Foster</marc:arc>
</metadata>
```

RDF/XML

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

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

```
  xmlns:marc="http://www.loc.gov/marc.relators/">
  <rdf:Description rdf:about="">
    <dc:title>Reichstag</dc:title>
    <marc:arc>Sir Norman Foster</marc:arc>
  </rdf:Description>
</rdf:RDF>
```

Guidelines for using Agent Roles in Dublin Core (Draft)

1/23/04, dih
2/12/04, rsg
2/16/04, dih
2/23/04, tb (converted to plain text only)

Function of roles and history of approval by DCMI

The need to express a role for the "Agent" elements (Creator, Contributor, and, to a lesser extent, Publisher) in the Dublin Core element Set has been expressed for many years and there have been a number of ideas for how to handle this. A role is a term that further refines the contribution of the agent to the resource described; an example is "illustrator" might be a role associated with the element Contributor when that person provided, for example, illustrations to the resource rather than being entirely responsible for all aspects of the intellectual content.

The DCMI Usage Board has discussed the issue several times and agreed that the role values are properly element refinements. At its meeting in Florence in Oct. 2002, the Board generally agreed that deprecating Creator and Publisher in favor of Contributor should be considered, since the role values Creator and Publisher could be used to refine Contributor if needed. Because of negative response to this sort of change in the Dublin Core element set, the DCMI Usage Board decided to recommend role refinements only to Contributor, but to leave Creator, Contributor and Publisher as separate elements.

Because a standardized, widely adopted list of roles already existed in the MARC Code List for Relators, it was recommended early in these discussions that DCMI not develop its own. In early 2000 a subset of the MARC relators list was circulated for approval to include 10 generally applicable roles out of the approximately 150 MARC relator terms. Because of the need to consider other aspects of the "Agent" issue, at that time the proposal was deferred. Discussion initially in the DC-Libraries Working Group (which prepared an application profile for use in libraries that included the relator terms) and later in the DCMI Usage Board revealed that consensus was that users not be constrained by a small list of elements and that the entire list of relator terms should be available given the difficulty of anticipating needs in various domains.

The Library of Congress has prepared an RDF expression of the MARC relator list, to be used in conjunction with the Dublin Core element Contributor. This is available at: http://www.loc.gov/marc/dc/marcrelcodes.rdf (note: this is not yet a persistent URI but Rebecca will provide one!).

The MARC Relator list, what it is and how it's structured

The MARC Code List for Relators was developed for use in MARC 21 bibliographic records to express the relationship between a name and a work. The list includes both role terms and three-character codes that represent those terms. The terms were only included on the list if the name and its associated role were considered important enough to include on a bibliographic record as a contributor. Thus, all terms

represent possible contributions to a resource. The Library of Congress is the maintenance agency for this list and regularly adds new ones when the need is expressed and documented.

The MARC Relator list includes three-character alphabetic codes to be used to identify roles. These are to be considered synonyms for the term they represent. In addition the list provides definitions for use of the term/code. In the RDF representation, the codes are tokens to be used for the term and are part of the URI. In some cases unused terms refer to used terms; these are included in the RDF representation as a note (dc:description). All terms are declared as refinements of dc:contributor.

Using Roles with Contributor

In order to minimize the possibility of confusion, the Usage Board has authorized the use of roles only with the element Contributor. Because Creator is in some sense a role elevated to a position at the level of element, the Board's position is that it makes no sense to provide yet another level of roles beneath that particular element.

Because Publisher and Creator each appears on the list of MARC Relator terms, and the entire list is available to be used as roles, it is acceptable to use the role of Publisher or Creator instead of the element within an implementation. If marc:creator is used instead of dc:creator, the two terms should probably not be used together in the same instance.

Because roles are to be used only with the Contributor element, appropriate "Dumb Down" of all agent refinements expressed as roles will be to Contributor. Given this, implementations may choose (preferably within the context of an application profile), to specify explicitly whether the MARC relator terms of publisher or creator may be used, based on the fact that the distinction could be retained in Simple Dublin Core if the Element level term is retained for those particular roles.

Using Roles in XML and other Schemas

Because the maintenance of the MARC Relator list will remain with the Library of Congress, the namespace of the roles will be established by LC and will not be a DCMI namespace. Thus, schemas will need to include the MARC relator namespace in order to properly express role terms. See the document Guidelines for Implementing Dublin Core in XML <http://dublincore.org/documents/dc-xml-guidelines/> for specific information on using non-DCMI namespaces.

Terms not on the MARC Relators list

The MARC Relator list has been developed over many years to meet a wide variety of needs. New terms are added on the basis of need, and LC has expressed willingness to continue to expand the list upon request. Implementers also have the option to create and expose alternative vocabularies for the expression of other kinds of roles not reflected in the MARC Relator list.

Managing the Use of Role in an Implementation

The MARC Relator list is large, including approximately 150

separate terms for various roles. Some of the roles on the list were created for specific domains, and would be of little use in other communities. It might therefore be useful for implementations to declare a subset of the role vocabulary as relevant to their specific goals, preferably by way of a formal application profile.

Topic: Dublin Core and OAI
Modified: 2004-02-25 15:44, Wednesday
Maintainer: Tom Baker
Latest version: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/oai/
See also: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/

Shepherd: Tom Baker

SUMMARY

We should discuss

-- the use of OAI-PMH to exchange metadata about "complex objects" (such as MPEG21 DIDLs), as the argument is sometimes made that it is inappropriate to use simple metadata when the objects are complex;

-- whether the Usage Board might play a useful role in clarifying some of the issues identified below;

-- more specifically, whether the Usage Board would even consider evaluating proposals for extensions to the DCMI Type Vocabulary, such as (hypothetically) "Person".

BACKGROUND

Report on breakout session: "The relationship between OAI-PMH and Dublin Core: Required, recommended or other?" [1]

OAI3 workshop, CERN, Geneva, 13 February 2004

Background to this session was a discussion last summer on the OAI mailing list (see [2], appended in plain text below, and the ensuing discussion thread).

The breakout session broke the problem down into several separate issues: 1) the inappropriateness or misuse of DC to describe non-document-like objects, 2) the low quality of many harvested metadata records in DC, 3) the question of whether DC elements were too non-specific to be useful, and 4) the observation that the effort of supplying DC for OAI-PMH compliance has -- rightly or wrongly -- discouraged some data providers from supplying richer metadata as well.

OAI-PMH already supports the harvesting of richer metadata in addition to the mandatory DC, so the issue was only whether the DC records should be made optional. In retrospect, it seemed like we were able in the alloted two hours to cover the main issues fairly well, though it would perhaps have been helpful to spend more time on the issue of Qualified as opposed to Simple Dublin Core.

Taking the issues in order:

1. Using DC in inappropriate ways. Some data providers have tried to use the fifteen elements to describe entities such as "people" and "places". I suggested that for each

major type of information described in the OAI context, guidelines be formulated to specify which elements should most appropriately be used in which ways. For example, a "person" might have a Type ("person") and an Identifier, and nothing else. Everyone seemed to agree that knowing that a resource was a "person" as opposed to a "physical object" or an "event" would be helpful -- or at any rate better than nothing! -- in assessing the utility of a resource.

2. Quality of metadata content. Junky records make junky indexes, and in many contexts (such as NSDL), the quality of metadata varies. Two points were made in response. Firstly, it was pointed out that the problem of bad metadata is not inherent to Dublin Core, but more general. As always, addressing the problem implies better tools and interfaces, partitioned workflows for metadata creation and quality control, and better guidelines for the use of DC in specific communities. Secondly, it was pointed out that DC is used not just for indexing and searching, but for listing result sets for browsing. Given the difficulty of extracting even basic information such as Titles and Authors on the fly from a broad diversity of metadata structures, dispensing with Dublin Core would mean that useful basic information about resources could often not be displayed at all.

3. Non-specific metadata semantics. In response to the complaint that DC elements are too imprecise to be useful, it was pointed out that is a problem inherent to the limitations of Simple DC (uncontextualized string values) as opposed to Qualified DC. It would seem that harvesters are particularly concerned about dc:identifier and dc:subject, where they are currently often reduced to crudely guessing the context of a subject heading or identifier. For dc:identifier in particular, it was suggested that this be addressed through more strongly encouraging the use of URIs. Beyond that, it was suggested that DC be constrained in usage guidelines for OAI as a whole -- i.e., in an OAI application profile of Dublin Core. However, it was pointed out in response that attempts to make the use of elements more specific in order to solve the problem for some inevitably seem to create problems for others -- a "classic" dilemma of interoperability.

4. Using DC instead of richer formats. In the OAI community, it has been observed that data providers focus on meeting the Dublin Core requirement only -- instead of also providing richer metadata (such as IEEE/LOM, MARC, Academic Metadata Format, OLAC, and RDF combinations of multiple metadata vocabularies). This apparently happens even when the provider already has richer metadata which it _could_ export in parallel to DC -- something which OAI-PMH currently supports. In other words, people with richer metadata are dumbing down to DC, while people with no metadata create DC records and then stop, even if DC is not ideal.

In response, it was suggested that OAI proceed on two fronts:
-- by actively promoting the use (or re-use) of richer
metadata alongside DC;
-- that future versions of the protocol support the indication
of a preference for the richer formats -- for example, by
pointing to preferred metadata from the "about" container
of the required DC metadata.

Summary of the group mood:

-- Keep Dublin Core mandatory
-- Encourage the use of richer formats parallel to DC
-- Support the indication of "preferred" formats
-- Constrain insufficiently specific elements with usage
recommendations ("use a URI with dc:identifier")
-- Explain how to use DC with non-document-like resources
("for people, just use Identifier and Type")

Open issues:
-- One important constraint is that no solution (e.g.,
changing future versions of OAI-PMH) may invalidate
current implementations of OAI-PMH
-- There should be some clarification of the role of
empty DC records which are (in fact) compliant.

REFERENCES

[1] http://www.openarchives.org/pipermail/oai-implementers/2003-August/000945.html
[2] http://info.web.cern.ch/info/OAIP/Breakout.html#group2

Date: Mon, 4 Aug 2003 15:49:17 -0400
From: "Carl Lagoze" <lagoze@cs.cornell.edu>
To: <oai-implementers@oaisrv.nsdl.cornell.edu>

Dublin Core has been the mandated metadata format in OAI-PMH since the initial release of the protocol. The purpose of this requirement was to promote interoperability among data providers. It was the subject of considerable discussion in the formulation of both the 1.0 and 2.0 specifications and we think that it is time to reexamine this requirement in light of two factors:

1. There is increasing interest in using the protocol for applications other than sharing metadata to promote resource discovery [1][2]. Dublin Core is targeted mainly as metadata for resource discovery and is, therefore, inappropriate for these other applications. It might make sense to loosen the Dublin Core requirement to a recommendation, and thus reduce any barrier to these broader applications.

2. The linkage between Dublin Core and OAI-PMH has been over-emphasized at the expense of the utility of OAI-PMH for dissemination of richer, and perhaps more useful, structured data. In some cases data providers with richer metadata (e.g., MARC, IEEE LOM) have been forced by the requirement to dumb-down rich metadata to Dublin Core and have then failed to provide the original metadata. As a result, the community

looses the benefits of rich data and is left with the reduced semantics of Dublin Core.

We need to choose between the competing goals of protocol stability and generalization. Although removing the Dublin Core requirement would not negatively impact existing or future data providers, it may impact service providers whose applications depend on the existence of a uniform metadata format.

We would like to open this subject for community discussion. While the technical aspects of this change are minimal it does have considerable political impact. Please give your feedback on the following proposal:

1. Change the Dublin Core requirement to a recommendation.
2. Leave oai_dc as a reserved metadataPrefix
3. Move the oai_dc part of protocol document to Implementation Guidelines

We invite members of this list to contribute their thoughts on this.

[1] http://www.dlib.org/dlib/july03/young/07young.html
[2] http://agenda.cern.ch/askArchive.php?base=agenda&categ=a02333&id=a02333s5t5/transparencies

Carl Lagoze
Michael Nelson
Herbert Van de Sompel
Simeon Warner

Date: Tue, 17 Feb 2004 18:48:16 +0100
From: Roland Schwaenzl <Roland.Schwaenzl@MATHEMATIK.UNI-OSNABRUECK.DE>

I'm a bit lunified about the criticism concerning DC.

In my view the problem described is an artefact of XML Schema binding - in the way it is quite often done. [...and to my understanding is done by OAI as well].

Not at all we should accept the criticism in DC15/plainLiteralValue metadata.

Low effort seldom results in sustained big gain.

DCMI since years is developing more expressive properties.

The problem is, that explicit listing of all properties (and classes) allowed in an XML (!) Schema does not scale well in this respect - additionally one still has the problem of relating XML Elements defined in an XML Schema with invariant URI's.

At Seattle my core assertion the need to develop the typed object aspect of metadata further within DCMI.

In Math-Net since several years we use typed objects along with DC (of course we created classes for persons, organizations and ...) By far we are not the only initiative which did so in the meantime.

I don't know how often i pointed to the problem of records, which use DC15/plainLiteralValue syntactically but in reality put codes from classification schemes into the subject field.

Of course the receiving application will judge such metadata as low quality.

Topic: Old business - etc
Modified: 2004-02-25 15:44, Wednesday
Maintainer: Tom Baker
Latest version: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/etc-old/
See also: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/

ITHACA/SEATTLE ACTION (Directorate)
Talk with ISO and NISO re posting copies of standards on DCMI website.
We need to check with NISO and ISO what the rules are that they have for 'Maintenance Agencies'
-- do we get the right to post the final text of these standards on our Web site?
-- how can we make changes and feed them back into the standard (e.g. can we fast-track the capitalization issue)?

ITHACA ACTION (Stuart)
The UB process document be amended to clarify that change requests from Working Groups should not combine a number of unrelated proposals.

ITHACA ACTION (Diane and Stuart)
Work on the DCMI Type vocabulary to move examples in definitions into a comment field.

SEATTLE Action Item 7: Diane to remove references in "Using DC" to using a label within relation if a refinement is not being used.

ITHACA ACTION ITEM Andy + Andrew: Revise DCSV Specification and various DCSV scheme specifications after approval of abstract model recommendation.

Topic: Usage Board issues
Modified: 2004-02-25 15:44, Wednesday
Maintainer: Tom Baker
Latest version: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/usageboard/
See also: http://www.bi.fhg.de/People/Thomas.Baker/ISSUES/

Shepherd: Tom Baker

SUMMARY

1. Voting in the Usage Board

   The DCMI Directorate has been working with the Board of Trustees on a set of by-laws for DCMI. One aspect of the draft -- on the request of the trustees -- will be to formulate a coherent approach to voting that cross-cuts different types of committees, including the Usage Board. Tom will circulate copies of the section of the bylaws dealing with the Usage Board for discussion and feedback.

2. Recruitment

   Tom will circulate an excerpt from the draft bylaws in which "liaisons" are defined (based on the proposal by Diane and Rebecca).

3. Next meeting in Shanghai

   http://www.bi.fhg.de/People/Thomas.Baker/CACHE/DC-2004-call.txt

   October 2004

Su	Mo	Tu	We	Th	Fr	Sa
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

CALL FOR PAPERS

DC-2004
International Conference on Dublin Core
and Metadata Applications

11-14 October 2004
Shanghai, China
http://dc2004.library.sh.cn/

Host: Shanghai Library
- http://www.library.sh.cn/new-eng/
Sponsors: Library of Chinese Academy of Sciences
- http://www.las.ac.cn/
National Science and Technology Library of China
- http://www.nstl.gov.cn/
Dublin Core Metadata Initiative
- http://dublincore.org/

Metadata based on standards such as Dublin Core is a key component of information environments from scientific repositories to corporate intranets and from business and publishing to education and e-government.

DC-2004 -- the fourth in a series of conferences previously held in Tokyo, Florence, and Seattle -- will examine a broad range of metadata applications, especially with a view towards improving interoperability across boundaries of language, culture, and communities of practice. Tutorials will provide an introduction to metadata for non-experts.

In conjunction with DC-2004, the Dublin Core Metadata Initiative will hold technical working-group meetings and Shanghai Library will host the 2004 Shanghai International Library Forum (http://www.libnet.sh.cn/silf2004/).

The Conference track of DC-2004 invites submissions of papers in the following areas:

-- Conceptual models for metadata
-- Interoperability among metadata systems and standards
-- Enterprise metadata
-- Globalization, localization, and multilinguality of metadata
-- Metadata for education, e-Government, and geospatial applications
-- Institutional Repositories and metadata harvesting
-- Search engines and metadata
-- Systems and tools for metadata applications
-- Metadata for cultural heritage and long-term preservation
-- Surveys, case studies, and novel activities based on metadata
-- Metadata registries and registry services
-- Metadata standardization
-- Ontologies and Semantic Web
-- Knowledge management

Paper Categories

Regular Papers (8 to 10 pages) describe original work in detail.

Short papers (2 to 4 pages) and posters describe a specific model, application, or activity in a concise format.

Author guidelines and submission details can be found at

http://dc2004.library.sh.cn/papers/.

All submissions will be peer-reviewed by the program committee and published both in print and electronically in the conference proceedings.

All accepted papers must be presented at the conference by at least one of their authors.

Deadlines

Paper submission	1 May 2004
Poster submission	15 May 2004
Acceptance notification	1 June 2004
Camera-ready copy due	1 July 2004

Previous Dublin Core conferences

DC-2001, Tokyo	http://www.nii.ac.jp/dc2001/
DC-2002, Florence	http://www.bncf.net/dc2002/
DC-2003, Seattle	http://dc2003.ischool.washington.edu/

Organization and chairs

Program Co-chairs	Liu Wei (Shanghai Library) Thomas Baker (Fraunhofer, Germany)
Workshop Chair	Makx Dekkers (DCMI)