Title: DCMI Usage Board - Singapore - 24-25 August 2007
Identifier: http://stage.dublincore.org/usage/meetings/2007/08/singapore/.html/ index.html
Source: http://stage.dublincore.org/usage/meetings/2007/08/singapore/index.txt
Created: 2007-06-27

Saturday

1. Changes to terms of the DCTERMS namespace (Diane)
   http://stage.dublincore.org/usage/meetings/2007/08/singapore/.html/dcterms-changes.html - topic page
   http://dublincore.org/usage/public-comment/2007/07/dcterms-changes/

2. Structure of DCMI Metadata Terms document (Tom)
   http://dublincore.org/usageboardwiki/DcmiTermsOutline?action=print

3. Review of application profiles (Joe, Stuart)
   http://stage.dublincore.org/usage/meetings/2007/08/singapore/.html/profile-review.html - topic page
   http://dublincore.org/usageboardwiki/ProfileReviewCriteria?action=print - old criteria
   http://dublincore.org/usageboardwiki/ProfileReviewCriteriaNew?action=print - new (Aug 2007) criteria
   http://stage.dublincore.org/usage/meetings/2007/08/singapore/pdf-term-decision.pdf
   -- dublincore.org/architecturewiki/TermDecisionTree?action=print
   http://dublincore.org/usageboardwiki/SESandVES?action=print
   http://stage.dublincore.org/usage/meetings/2007/08/singapore/.html/CollectionsProfileReview.html

-- Background documents on process
   http://dublincore.org/documents/approval/
   http://dublincore.org/usage/documents/process/
   http://dublincore.org/usage/meetings/2007/03/barcelona/Process_Doc_Revisions.txt

4. FOAF (Andrew) - 60"
   http://stage.dublincore.org/usage/meetings/2007/08/singapore/.html/foaf.html - topic page
   http://dublincore.org/groups/agents/agentFRdraft2-2.html
   http://dublincore.org/agentswiki/FoafReview?action=print
   http://stage.dublincore.org/usage/meetings/2007/08/singapore/Eprints-excerpt.pdf
   http://rdfweb.org/mt/foaflog/archives/2003/07/10/12.05.33/

5. Changes to term declarations in RDF (Tom)
   http://stage.dublincore.org/usage/meetings/2007/08/singapore/.html/rdf-declaration-changes.html - topic page

---

Sunday

6. Domains and ranges (Akira)
   http://stage.dublincore.org/usage/meetings/2007/08/singapore/.html/domains-ranges.html - topic page
   http://stage.dublincore.org/usage/meetings/2007/08/singapore/pdf-domains.pdf
   -- dublincore.org/documents/2007/07/02/domain-range/
   http://dublincore.org/usageboardwiki/RangesIssues?action=print
   http://stage.dublincore.org/usage/meetings/2007/08/singapore/domains-digest1.txt
   http://stage.dublincore.org/usage/meetings/2007/08/singapore/domains-digest2.txt

7. Description Set Profile model and wiki syntax (Mikael as guest)
   http://stage.dublincore.org/usage/meetings/2007/08/singapore/.html/description-set-profile.html - topic page
   http://stage.dublincore.org/usage/meetings/2007/08/singapore/pdf-dcap-model.pdf
   http://dublincore.org/architecturewiki/DescriptionSetProfile?action=print
   http://stage.dublincore.org/usage/meetings/2007/08/singapore/pdf-eprints.pdf
   -- knowware.nada.kth.se/DCWiki/EprintsApplicationProfile?action=print
   http://kmr.nada.kth.se/wiki/Main/MoinMoinWikiFormatSuggestion1?action=print

8. Application Profile of Simple Dublin Core (Tom, Mikael)
   http://knowware.nada.kth.se/DCWiki/SimpleDublinCore?action=print

Title: DCMI Usage Board - Singapore - 24-25 August 2007
Identifier: http://stage.dublincore.org/usage/meetings/2007/08/singapore/.html/ index.html
Source: http://stage.dublincore.org/usage/meetings/2007/08/singapore/index.txt
Created: 2007-06-27

Saturday

1. Changes to terms of the DCTERMS namespace (Diane)
   http://stage.dublincore.org/usage/meetings/2007/08/singapore/.html/dcterms-changes.html - topic page
   http://dublincore.org/usage/public-comment/2007/07/dcterms-changes/

2. Structure of DCMI Metadata Terms document (Tom)
   http://dublincore.org/usageboardwiki/DcmiTermsOutline?action=print

3. Review of application profiles (Joe, Stuart)
   http://stage.dublincore.org/usage/meetings/2007/08/singapore/.html/profile-review.html - topic page
   http://dublincore.org/usageboardwiki/ProfileReviewCriteria?action=print - old criteria
   http://dublincore.org/usageboardwiki/ProfileReviewCriteriaNew?action=print - new (Aug 2007) criteria
   http://stage.dublincore.org/usage/meetings/2007/08/singapore/pdf-term-decision.pdf
   -- dublincore.org/architecturewiki/TermDecisionTree?action=print
   http://dublincore.org/usageboardwiki/SESandVES?action=print
   http://stage.dublincore.org/usage/meetings/2007/08/singapore/.html/CollectionsProfileReview.html

-- Background documents on process
   http://dublincore.org/documents/approval/
   http://dublincore.org/usage/documents/process/
   http://dublincore.org/usage/meetings/2007/03/barcelona/Process_Doc_Revisions.txt

4. FOAF (Andrew) - 60"
   http://stage.dublincore.org/usage/meetings/2007/08/singapore/.html/foaf.html - topic page
   http://dublincore.org/groups/agents/agentFRdraft2-2.html
   http://dublincore.org/agentswiki/FoafReview?action=print
   http://stage.dublincore.org/usage/meetings/2007/08/singapore/Eprints-excerpt.pdf
   http://rdfweb.org/mt/foaflog/archives/2003/07/10/12.05.33/

5. Changes to term declarations in RDF (Tom)
   http://stage.dublincore.org/usage/meetings/2007/08/singapore/.html/rdf-declaration-changes.html - topic page

---

Sunday

6. Domains and ranges (Akira)
   http://stage.dublincore.org/usage/meetings/2007/08/singapore/.html/domains-ranges.html - topic page
   http://stage.dublincore.org/usage/meetings/2007/08/singapore/pdf-domains.pdf
   -- dublincore.org/documents/2007/07/02/domain-range/
   http://dublincore.org/usageboardwiki/RangesIssues?action=print
   http://stage.dublincore.org/usage/meetings/2007/08/singapore/domains-digest1.txt
   http://stage.dublincore.org/usage/meetings/2007/08/singapore/domains-digest2.txt

7. Description Set Profile model and wiki syntax (Mikael as guest)
   http://stage.dublincore.org/usage/meetings/2007/08/singapore/.html/description-set-profile.html - topic page
   http://stage.dublincore.org/usage/meetings/2007/08/singapore/pdf-dcap-model.pdf
   http://dublincore.org/architecturewiki/DescriptionSetProfile?action=print
   http://stage.dublincore.org/usage/meetings/2007/08/singapore/pdf-eprints.pdf
   -- knowware.nada.kth.se/DCWiki/EprintsApplicationProfile?action=print
   http://kmr.nada.kth.se/wiki/Main/MoinMoinWikiFormatSuggestion1?action=print

8. Application Profile of Simple Dublin Core (Tom, Mikael)
   http://knowware.nada.kth.se/DCWiki/SimpleDublinCore?action=print

Title: Changes to terms of the DCTERMS namespace
Identifier: http://stage.dublincore.org/usage/meetings/2007/08/singapore/.html/dcterms-changes.html
Created: 2007-07-19

Changes to terms of the DCTERMS namespace (Diane)

"Revisions to DCMI Metadata Terms" were published for Public Comment on Monday, 2 July [1]. These will be discussed and finalized in Singapore. Sources were:

-- Barcelona decisions [2]
-- Changes made between DCTERMS-2006 [6] and NISO Z39-85-2007 [5]

[1] http://dublincore.org/usage/public-comment/2007/07/dcterms-changes/
[2] http://dublincore.org/usageboardwiki/DCTermsChangesActions
[3] http://dublincore.org/usage/meetings/2007/03/barcelona/2007-03-16.ub-agenda-barcelona.pdf
[4] http://www.niso.org/standards/resources/Z39-85-2007.pdf
[5] http://dublincore.org/documents/2006/12/18/dcmi-terms/

Revisions to DCMI Metadata Terms

Creator: Thomas Baker
DCMI
Date Issued: 2007-07-02
Identifier: http://dublincore.org/usage/public-comment/documents/2007/07/dcterms-changes/
Description of Document: This set of proposed editorial clarifications to descriptions of DCMI metadata terms is proposed by the DCMI Usage Board for Public Comment.

Table of contents

1. About the proposed revisions
2. Types of revisions
3. Revisions, term by term
References

1. About the proposed revisions

The document proposes a set of editorial revisions to terms in the set of DCMI Metadata Terms [DCTERMS]. The proposed changes are summarized in Section 2 and detailed, term by term, in Section 3. This proposal should be considered in conjunction with a separate but related proposal, "Domains and Ranges for DCMI Properties", which proposes a set of classes for assignment as domains and ranges of DCMI properties [DOMAINS].

This document follows on an earlier revision of terms in the Dublin Core Metadata Element Set (DCMES) [DCMES-CHANGES], which was undertaken in order to clarify intended semantics and bring the wording of definitions and usage comments into line with the language of the DCMI Abstract Model [DCAM].

After a Public Comment period, the changes described here will be finalized by the DCMI Usage Board, then published in revised versions of the Web documents "DCMI Metadata Terms" [DCTERMS] and "DCMI Metadata Terms: A complete historical record" [DCTERMS-HISTORY].

All of the changes proposed here have been evaluated by the DCMI Usage Board in light of the DCMI Namespace Policy [NAMESPACE]. The namespace policy says that DCMI terms are identified using Uniform Resource Identifiers (URIs). In accordance with the principle that distinct URIs should be assigned to distinct resources, the policy sets limits on the range of editorial changes that may allowably be made to the official labels, definitions, and usage comments associated with DCMI terms. By policy, any changes of meaning judged "likely to have a substantial impact on either machine processing of DCMI terms or the functional semantics of the terms" must trigger the creation of a new, distinct term

with a new, distinct URI.

In the opinion of the Usage Board, the changes described in this document are unlikely to have a substantial impact on either machine processing of DCMI terms or the functional semantics of the terms -- i.e., they fall within the allowable range of editorial change. The changes constitute clarifications of term semantics in light of improved theoretical understanding, user feedback, and implementation experience. They are intended to provide the final twist of the lens that brings intended meanings more sharply into focus.

2. Types of revisions

Free-standing descriptive labels

When terms such as dcterms:available were first coined as "Dublin Core qualifiers" in 2000 [QUALIFIERS], it was not yet entirely clear whether qualifiers were intended to be used like adjectives -- i.e., in conjunction with the Dublin Core elements they modified -- or as elements in their own right. The labels assigned at the time, such as "Available", suggest the former, but DCMI grammatical principles, and later the Abstract Model, clearly evolved to the latter. The terms dcterms:alternative, dcterms:available, dcterms:created, dcterms:issued, dcterms:modified, dcterms:valid, dcterms:spatial, and dcterms:temporal have been re-named with free-standing names such as "Date created" (instead of just "Created") and "Alternative Title" instead of just "Alternative").

The phrase "content of the resource"

As described in [DCMES-CHANGES], wordings which distinguish the "content of" a resource have been revised to refer directly to a resource. In the current batch, this change is relevant to the definitions of: dcterms:abstract, dcterms:tableOfContents, dcterms:isFormatOf, dcterms:hasFormat, dcterms:spatial, dcterms:temporal, dcterms:DCMIType, and dcterms:requires.

The phrase "reference to a resource"

Definitions which specify encoding practice with phrases such as "reference to a resource" have been revised as described in [DCMES-CHANGES]. This change is relevant to the definition of dcterms:conformsTo. Note, however, that the definition of dcterms:bibliographicCitation remains "A bibliographic reference for the resource".

The phrase "use a controlled vocabulary"

It is now generally recommended that controlled vocabularies be used in metadata, so specific usage advice along these lines has been removed from the comments dcterms:accrualMethod, dcterms:accrualPeriodicity, and dcterms:accrualPolicy.

Types of "encoding schemes"

The revised DCMI Abstract Model [DCAM] draws a clear distinction between Vocabulary Encoding Schemes and Syntax Encoding Schemes. Existing terms hitherto classified as "encoding schemes" have been designated in this batch of revisions, more specifically, as Vocabulary Encoding Schemes (dcterms:DCMIType, dcterms:DDC, dcterms:IMT, dcterms:LCC, dcterms:LCSH, dcterms:MESH, dcterms:NLM, dcterms:RFC4646, dcterms:TGN, and dcterms:UDC) or Syntax Encoding Schemes (dcterms:Box, dcterms:ISO3166, dcterms:ISO639-2, dcterms:Period, dcterms:Point, dcterms:RFC1766, dcterms:RFC3066, dcterms:URI, and dcterms:W3CDTF).

Encoding scheme definitions

In accordance with the differentiation of encoding schemes into Vocabulary Encoding Schemes and Syntax Encoding Schemes, the Usage Board has revised their definitions to make clear how the encoding

schemes are being seen and interpreted from a modeling point of view:

> as a set of concepts: dcterms:IMT, dcterms:LCSH, dcterms:MESH, dcterms:DDC, dcterms:LCC, dcterms:UDC, dcterms:DCMIType, dcterms:NLMas a set of codes: dcterms:ISO639-2, dcterms:ISO3166as a set of tags: dcterms:RFC1766as a set of identifiers: dcterms:URIas a set of regions in space: dcterms:Boxas a set of places: dcterms:TGNas a set of time intervals: dcterms:Periodas a set of dates and times: dcterms:W3CDTF

A thorough discussion of these interpretations is beyond the scope of this document and will be undertaken by the Usage Board in a separate note.

Definitions versus Comments

Parts of some term Definitions have been moved into Comments, as with dcterms:isVersionOf, dcterms:dateAccepted, and dcterms:dateSubmitted.

Relation of a property to a "described resource"

Some definitions have revised to make clear how a property relates to the "described resource" (dcterms:instructionalMethod, dcterms:isVersionOf, dcterms:isReplacedBy, dcterms:replaces, dcterms:isRequiredBy, dcterms:requires, dcterms:isPartOf, dcterms:hasPart, dcterms:isReferencedBy, dcterms:references, dcterms:isFormatOf, and dcterms:hasFormat).

References to IETF Requests for Comment (RFCs)

Some term declarations make reference to "Request for Comment" (RFC) standards maintained by The Internet Engineering Task Force (IETF). When RFC standards are updated, their numbers change, and the newer standard (with a higher number) obsoletes the older standard. In this batch of revisions, references to RFCs have been updated:

- for dcterms:URI, RFC 3986 has obsoleted RFC 2396
- for dcterms:language, RFC 4646 has obsoleted RFC 3066

DCMI encoding schemes that designate RFCs

Existing encoding schemes declared by DCMI in reference to specific RFCs -- dcterms:RFC1766 and dcterms:RFC3066 -- are interpreted in this batch of proposed changes as Syntax Encoding Schemes. Since RFC 3066 has been obsoleted by RFC 4646, however, this batch of revisions includes the proposal for a new DCMI encoding scheme, dcterms:RFC4646, which is interpreted to be a Vocabulary Encoding Scheme.

Revisions in response to the five-year review for NISO Z39.85-2007

A five-year review of ANSI/NISO Z39.85 (The Dublin Core Metadata Element Set) was held, resulting in the revised standard ANSI/NISO Z39.85-2007 in May 2007 [NISOZ39-85]. This batch of revisions includes minor changes of wording undertaken in response to comments (dcterms:source, dcterms:subject, dcterms:language, and dcterms:coverage). In response to the comment that the Comment for dcterms:title was potentially misleading, the Usage Board decided to delete the comment altogether.

The statement "qualifies:"

When "Dublin Core qualifiers" were first published in 2000 [QUALIFIERS], encoding schemes were associated with specific elements -- e.g., dcterms:LCSH with dc:subject. In light of subsequent work on the DCMI model, however, the association between a property and a Vocabulary Encoding Scheme or Syntax Encoding Scheme is seen as something more appropriately declared in an application profile than in the formal declaration of a vocabulary. "Qualifies:" statements have therefore been removed from the

term descriptions for dcterms:DDC, dcterms:LCC, dcterms:UDC, dcterms:DCMIType, dcterms:IMT, dcterms:ISO639-2, dcterms:RFC1766, dcterms:URI, dcterms:Point, dcterms:ISO3166, dcterms:Box, dcterms:TGN, dcterms:Period, dcterms:W3CDTF, dcterms:RFC3066, dcterms:NLM, dcterms:LCSH, and dcterms:MESH. Methods for the formal expression of such an association in the RDF schemas published by DCMI were never finalized as a DCMI Recommendation [DCQ-RDF] and have since been superseded by new guidelines for expressing Dublin Core metadata in RDF [DC-RDF], as discussed in [DC-RDF-NOTES].

SubjectScheme and other "undeclared" members of the TERMS namespace

It is worth noting that since March 2000, seven classes with URIs based on http://purl.org/dc/terms/ (the "TERMS" namespace) have been used in RDF schemas published by DCMI. These classes were dcterms:DateScheme, dcterms:FormatScheme, dcterms:IdentifierScheme, dcterms:LanguageScheme, dcterms:SpatialScheme, dcterms:SubjectScheme, and dcterms:TypeScheme. These classes were added to the RDF expressions on the basis of "qualfies:" statements (see above) as part of an automatic routine for generating updated documentation for DCMI terms. At the time, the classes served to associate a set of encoding schemes with a particular property (e.g., to associate dcterms:LCSH with dc:subject). Since the terms were never officially declared (outside of their undocumented use in the RDF schemas), DCMI does not consider them to fall under the DCMI Namespace Policy [NAMESPACE]. With the removal of the "qualifies:" statements and the revision of guidelines for expressing Dublin Core metadata in RDF (see above), these undeclared classes will cease to be used in DCMI term declarations.

New terms related to the DCMI Abstract Model

As described in the revised DCMI Namespace Policy [NAMESPACE], a DCMI namespace has been created as the base URI for terms used in the DCMI Abstract Model (http://purl.org/dc/dcam/, here dcam:). Two new dcam: terms -- dcam:isMemberOf and dcam:VocabularyEncodingScheme -- have been created with the finalization of the revised DCMI Abstract Model as a DCMI Recommendation in June 2007 [DCAM].

Addition of domains and ranges for existing properties and creation of new properties

A separate document, "Domains and ranges for DCMI properties" [DOMAINS], proposes a vocabulary of classes for use as formal domains and ranges of DCMI properties. Domains and ranges specify -- in a form usable for inferencing -- what kind of described resources and value resources are associated with a given property. The assignment of formal domains and ranges makes the meanings implicit in natural-language definitions available for machine processing. So as not to affect the conformance of existing implementations of Simple Dublin Core in RDF, domains and ranges will not be specified for the fifteen properties of the Dublin Core Metadata Element Set (DCMES), Version 1.1, which are identified by URIs in the DCMI namespace http://purl.org/dc/elements/1.1/ . Rather, the document proposes to define fifteen new properties, which will be identified by URIs in the DCMI namespace http://purl.org/dc/terms/ . Each of the fifteen new properties is to be a subproperty of the corresponding property of the DCMES, and is to be assigned domain and range as outlined in the proposal.

Subproperty relations between terms in "the Dublin Core"

The original thirteen (later fifteen) elements of "the Dublin Core" were originally intended to serve as a self-contained template for simple descriptions. It was soon noticed that some of the fifteen elements semantically fit within the scope of others, but a typological distinction between "elements" and "qualifiers" made it seem confusing to declare some elements to be subproperties of others. The DCMI Abstract Model no longer makes such a typological distinction, so the current batch of revisions includes assertions of dcterms:creator as a sub-property of dcterms:contributor and of dcterms:source as a sub-property of dcterms:relation. As with the assignment of domains and ranges, so as not to affect the conformance of existing implementations, these assertions are made only for the newly created properties, identified by URIs in the DCMI namespace http://purl.org/dc/terms/ . No such assertions are made for the existing properties of the DCMES.

Comments deleted

In addition to the Comment for dcterms:title, other Comments were found to be no longer useful and

simply deleted (dcterms:audience, dcterms:license dcterms:rightsHolder, dcterms:accrualMethod, dcterms:accrualPeriodicity, dcterms:accrualPolicy).

Special cases

Editorial revision has sometimes involved deciding among different possible interpretations (e.g., dcterms: educationLevel) and minor clarifications of wording too subtle to merit separate explanations (e.g., dcterms:accessRights). Section 3 lists old and new versions side-by-side so that readers may judge for themselves.

3. Revisions, term by term

http://purl.org/dc/terms/creator	
New subPropertyOf	http://purl.org/dc/terms/contributor

http://purl.org/dc/terms/source	
Current definition	A resource from which the described resource is derived.
New definition	A related resource from which the described resource is derived.
New subPropertyOf	http://purl.org/dc/terms/relation

http://purl.org/dc/elements/1.1/coverage	
Current comment	Spatial topic may be a named place or a location specified by its geographic coordinates. Temporal period may be a named period, date, or date range. A jurisdiction may be a named administrative entity or a geographic place to which the resource applies. Recommended best practice is to use a controlled vocabulary such as the Thesaurus of Geographic Names [TGN]. Where appropriate, named places or time periods can be used in preference to numeric identifiers such as sets of coordinates or date ranges.
New comment	Spatial topic and spatial applicability may be a named place or a location specified by its geographic coordinates. Temporal topic may be a named period, date, or date range. A jurisdiction may be a named administrative entity or a geographic place to which the resource applies. Recommended best practice is to use a controlled vocabulary such as the Thesaurus of Geographic Names [TGN]. Where appropriate, named places or time periods can be used in preference to numeric identifiers such as sets of coordinates or date ranges.

http://purl.org/dc/elements/1.1/language	
Current comment	Recommended best practice is to use a controlled vocabulary such as RFC 3066 [RFC3066].
Current see also	http://www.ietf.org/rfc/rfc3066.txt
New comment	Recommended best practice is to use a controlled vocabulary such as RFC 4646 [RFC4646].
New see also	http://www.ietf.org/rfc/rfc4646.txt

http://purl.org/dc/elements/1.1/subject	
Current comment	Typically, the topic will be represented using keywords, key phrases, or classification codes. Recommended best practice is to use a controlled vocabulary. To describe the spatial or temporal topic of the resource, use the Coverage element.
New comment	Typically, the subject will be represented using keywords, key phrases, or classification codes. Recommended best practice is to use a controlled vocabulary. To describe the spatial or temporal topic of the resource, use the Coverage element.

http://purl.org/dc/elements/1.1/title	
Current comment	Typically, A Title will be a name by which the resource is formally known.
New comment	[comment deleted]

http://purl.org/dc/terms/audience

Current comment	A class of entity may be determined by the creator or the publisher or by a third party.
Current type of term	http://dublincore.org/usage/documents/principles/#element
New comment	[comment deleted]
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property

http://purl.org/dc/terms/alternative	
Current label	Alternative
Current definition	Any form of the title used as a substitute or alternative to the formal title of the resource.
Current comment	This qualifier can include Title abbreviations as well as translations.
Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement
Current subPropertyOf	http://purl.org/dc/elements/1.1/title
New label	Alternative Title
New definition	An alternative name for the resource.
New comment	The distinction between titles and alternative titles is application-specific.
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
New subPropertyOf	http://purl.org/dc/terms/title

http://purl.org/dc/terms/tableOfContents	
Current definition	A list of subunits of the content of the resource.
Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement
Current subPropertyOf	http://purl.org/dc/elements/1.1/description
New definition	A list of subunits of the resource.
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
New subPropertyOf	http://purl.org/dc/terms/description

http://purl.org/dc/terms/abstract	
Current definition	A summary of the content of the resource.
Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement
Current subPropertyOf	http://purl.org/dc/elements/1.1/description
New definition	A summary of the resource.
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
New subPropertyOf	http://purl.org/dc/terms/description

http://purl.org/dc/terms/created	
Current label	Created
Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement
Current subPropertyOf	http://purl.org/dc/elements/1.1/date
New label	Date Created
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
New subPropertyOf	http://purl.org/dc/terms/date

http://purl.org/dc/terms/valid	
Current label	Valid
Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement
Current subPropertyOf	http://purl.org/dc/elements/1.1/date
New label	Date Valid
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
New subPropertyOf	http://purl.org/dc/terms/date

http://purl.org/dc/terms/available	
Current label	Available
Current definition	Date (often a range) that the resource will become or did become available.

Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement
Current subPropertyOf	http://purl.org/dc/elements/1.1/date
New label	Date Available
New definition	Date (often a range) that the resource became or will become available.
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
New subPropertyOf	http://purl.org/dc/terms/date

http://purl.org/dc/terms/issued	
Current label	Issued
Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement
Current subPropertyOf	http://purl.org/dc/elements/1.1/date
New label	Date Issued
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
New subPropertyOf	http://purl.org/dc/terms/date

http://purl.org/dc/terms/modified	
Current label	Modified
Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement
Current subPropertyOf	http://purl.org/dc/elements/1.1/date
New label	Date Modified
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
New subPropertyOf	http://purl.org/dc/terms/date

http://purl.org/dc/terms/extent	
Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement
Current subPropertyOf	http://purl.org/dc/elements/1.1/format
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
New subPropertyOf	http://purl.org/dc/terms/format

http://purl.org/dc/terms/medium	
Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement
Current subPropertyOf	http://purl.org/dc/elements/1.1/format
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
New subPropertyOf	http://purl.org/dc/terms/format

http://purl.org/dc/terms/isVersionOf	
Current definition	The described resource is a version, edition, or adaptation of the referenced resource. Changes in version imply substantive changes in content rather than differences in format.
Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement
Current subPropertyOf	http://purl.org/dc/elements/1.1/relation
New definition	A related resource of which the described resource is a version, edition, or adaptation.
New comment	Changes in version imply substantive changes in content rather than differences in format.
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
New subPropertyOf	http://purl.org/dc/terms/relation

http://purl.org/dc/terms/hasVersion	
Current definition	The described resource has a version, edition, or adaptation, namely, the referenced resource.
Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement
Current subPropertyOf	http://purl.org/dc/elements/1.1/relation
New definition	A related resource that is a version, edition, or adaptation of the described resource.

New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
New subPropertyOf	http://purl.org/dc/terms/relation

http://purl.org/dc/terms/isReplacedBy	
Current definition	The described resource is supplanted, displaced, or superseded by the referenced resource.
Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement
Current subPropertyOf	http://purl.org/dc/elements/1.1/relation
New definition	A related resource that supplants, displaces, or supersedes the described resource.
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
New subPropertyOf	http://purl.org/dc/terms/relation

http://purl.org/dc/terms/replaces	
Current definition	The described resource supplants, displaces, or supersedes the referenced resource.
Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement
Current subPropertyOf	http://purl.org/dc/elements/1.1/relation
New definition	A related resource that is supplanted, displaced, or superseded by the described resource.
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
New subPropertyOf	http://purl.org/dc/terms/relation

http://purl.org/dc/terms/isRequiredBy	
Current definition	The described resource is required by the referenced resource, either physically or logically.
Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement
Current subPropertyOf	http://purl.org/dc/elements/1.1/relation
New definition	A related resource that requires the described resource to support its function, delivery, or coherence.
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
New subPropertyOf	http://purl.org/dc/terms/relation

http://purl.org/dc/terms/requires	
Current definition	The described resource requires the referenced resource to support its function, delivery, or coherence of content.
Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement
Current subPropertyOf	http://purl.org/dc/elements/1.1/relation
New definition	A related resource that is required by the described resource to support its function, delivery, or coherence.
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
New subPropertyOf	http://purl.org/dc/terms/relation

http://purl.org/dc/terms/isPartOf	
Current definition	The described resource is a physical or logical part of the referenced resource.
Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement
Current subPropertyOf	http://purl.org/dc/elements/1.1/relation
New definition	A related resource in which the described resource is physically or logically included.
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
New subPropertyOf	http://purl.org/dc/terms/relation

http://purl.org/dc/terms/hasPart	
Current definition	The described resource includes the referenced resource either physically or logically.
Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement

Current subPropertyOf	http://purl.org/dc/elements/1.1/relation
New definition	A related resource that is included either physically or logically in the described resource.
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
New subPropertyOf	http://purl.org/dc/terms/relation

http://purl.org/dc/terms/isReferencedBy	
Current definition	The described resource is referenced, cited, or otherwise pointed to by the referenced resource.
Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement
Current subPropertyOf	http://purl.org/dc/elements/1.1/relation
New definition	A related resource that references, cites, or otherwise points to the described resource.
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
New subPropertyOf	http://purl.org/dc/terms/relation

http://purl.org/dc/terms/references	
Current definition	The described resource references, cites, or otherwise points to the referenced resource.
Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement
Current subPropertyOf	http://purl.org/dc/elements/1.1/relation
New definition	A related resource that is referenced, cited, or otherwise pointed to by the described resource.
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
New subPropertyOf	http://purl.org/dc/terms/relation

http://purl.org/dc/terms/isFormatOf	
Current definition	The described resource is the same intellectual content of the referenced resource, but presented in another format.
Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement
Current subPropertyOf	http://purl.org/dc/elements/1.1/relation
New definition	A related resource that is substantially the same as the described resource, but in another format.
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
New subPropertyOf	http://purl.org/dc/terms/relation

http://purl.org/dc/terms/hasFormat	
Current definition	The described resource pre-existed the referenced resource, which is essentially the same intellectual content presented in another format.
Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement
Current subPropertyOf	http://purl.org/dc/elements/1.1/relation
New definition	A related resource that is substantially the same as the pre-existing described resource, but in another format.
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
New subPropertyOf	http://purl.org/dc/terms/relation

http://purl.org/dc/terms/conformsTo	
Current definition	A reference to an established standard to which the resource conforms.
Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement
Current subPropertyOf	http://purl.org/dc/elements/1.1/relation
New definition	An established standard to which the resource conforms.
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
New subPropertyOf	http://purl.org/dc/terms/relation

http://purl.org/dc/terms/spatial

Current label	Spatial
Current definition	Spatial characteristics of the intellectual content of the resource.
Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement
Current subPropertyOf	http://purl.org/dc/elements/1.1/coverage
New label	Spatial Coverage
New definition	Spatial characteristics of the resource.
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
New subPropertyOf	http://purl.org/dc/terms/coverage

http://purl.org/dc/terms/temporal	
Current label	Temporal
Current definition	Temporal characteristics of the intellectual content of the resource.
Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement
Current subPropertyOf	http://purl.org/dc/elements/1.1/coverage
New label	Temporal Coverage
New definition	Temporal characteristics of the resource.
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
New subPropertyOf	http://purl.org/dc/terms/coverage

http://purl.org/dc/terms/mediator	
Current comment	The audiences for a resource are of two basic classes: (1) an ultimate beneficiary of the resource, and (2) frequently, an entity that mediates access to the resource. The mediator element refinement represents the second of these two classes.
Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement
New comment	In an educational context, a mediator might be a parent, teacher, teaching assistant, or care-giver.
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property

http://purl.org/dc/terms/dateAccepted	
Current definition	Date of acceptance of the resource (e.g. of thesis by university department, of article by journal, etc.).
Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement
Current subPropertyOf	http://purl.org/dc/elements/1.1/date
New definition	Date of acceptance of the resource.
New comment	Examples of resources to which a Date Accepted may be relevant are a thesis (accepted by a university department) or an article (accepted by a journal).
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
New subPropertyOf	http://purl.org/dc/terms/date

http://purl.org/dc/terms/dateCopyrighted	
Current definition	Date of a statement of copyright.
Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement
Current subPropertyOf	http://purl.org/dc/elements/1.1/date
New definition	Date of copyright.
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
New subPropertyOf	http://purl.org/dc/terms/date

http://purl.org/dc/terms/dateSubmitted	
Current definition	Date of submission of the resource (e.g. thesis, articles, etc.).
Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement
Current subPropertyOf	http://purl.org/dc/elements/1.1/date
New definition	Date of submission of the resource.
New comment	Examples of resources to which a Date Submitted may be relevant are a thesis (submitted to a university department) or an article (submitted to a journal).

2007-08-25 Usage Board meeting, Singapore

New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
New subPropertyOf	http://purl.org/dc/terms/date

http://purl.org/dc/terms/educationLevel	
Current definition	A general statement describing the education or training context. Alternatively, a more specific statement of the location of the audience in terms of its progression through an education or training context.
Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement
New definition	A class of entity, defined in terms of progression through an educational or training context, for whom the resource is intended.
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property

http://purl.org/dc/terms/accessRights	
Current comment	Access Rights may include information regarding access or restrictions based on privacy, security or other regulations.
Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement
Current subPropertyOf	http://purl.org/dc/elements/1.1/rights
New comment	Access Rights may include information regarding access or restrictions based on privacy, security or other policies.
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
New subPropertyOf	http://purl.org/dc/terms/rights

http://purl.org/dc/terms/bibliographicCitation	
Current comment	Recommended practice is to include sufficient bibliographic detail to identify the resource as unambiguously as possible, whether or not the citation is in a standard form.
Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement
Current subPropertyOf	http://purl.org/dc/elements/1.1/identifier
New comment	Recommended practice is to include sufficient bibliographic detail to identify the resource as unambiguously as possible.
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
New subPropertyOf	http://purl.org/dc/terms/identifier

http://purl.org/dc/terms/license	
Current comment	Recommended best practice is to identify the license using a URI. Examples of such licenses can be found at http://creativecommons.org/licenses/.
Current type of term	http://dublincore.org/usage/documents/principles/#element-refinement
Current subPropertyOf	http://purl.org/dc/elements/1.1/rights
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property
New subPropertyOf	http://purl.org/dc/terms/rights

http://purl.org/dc/terms/rightsHolder	
Current comment	Recommended best practice is to use the URI or name of the Rights Holder to indicate the entity.
Current type of term	http://dublincore.org/usage/documents/principles/#element
New comment	[comment deleted]
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property

http://purl.org/dc/terms/provenance	
Current type of term	http://dublincore.org/usage/documents/principles/#element
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property

http://purl.org/dc/terms/instructionalMethod	
Current definition	A process, used to engender knowledge, attitudes and skills, that the resource is designed to support.
Current type of term	http://dublincore.org/usage/documents/principles/#element

New definition	A process, used to engender knowledge, attitudes and skills, that the described resource is designed to support.
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property

http://purl.org/dc/terms/accrualMethod	
Current comment	Recommended best practice is to use a value from a controlled vocabulary.
Current type of term	http://dublincore.org/usage/documents/principles/#element
New comment	[comment deleted]
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property

http://purl.org/dc/terms/accrualPeriodicity	
Current comment	Recommended best practice is to use a value from a controlled vocabulary.
Current type of term	http://dublincore.org/usage/documents/principles/#element
New comment	[comment deleted]
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property

http://purl.org/dc/terms/accrualPolicy	
Current comment	Recommended best practice is to use a value from a controlled vocabulary.
Current type of term	http://dublincore.org/usage/documents/principles/#element
New comment	[comment deleted]
New type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property

http://purl.org/dc/terms/LCSH	
Current definition	Library of Congress Subject Headings
Current type of term	http://dublincore.org/usage/documents/principles/#encoding-scheme
Current "qualifies"	http://purl.org/dc/elements/1.1/subject
New definition	The set of concepts defined by Library of Congress Subject Headings.
New type of term	http://purl.org/dc/dcam/VocabularyEncodingScheme

http://purl.org/dc/terms/MESH	
Current definition	Medical Subject Headings
Current type of term	http://dublincore.org/usage/documents/principles/#encoding-scheme
Current "qualifies"	http://purl.org/dc/elements/1.1/subject
New definition	The set of concepts defined by Medical Subject Headings.
New type of term	http://purl.org/dc/dcam/VocabularyEncodingScheme

http://purl.org/dc/terms/DDC	
Current definition	Dewey Decimal Classification
Current type of term	http://dublincore.org/usage/documents/principles/#encoding-scheme
Current "qualifies"	http://purl.org/dc/elements/1.1/subject
New definition	The set of concepts defined by Dewey Decimal Classification.
New type of term	http://purl.org/dc/dcam/VocabularyEncodingScheme

http://purl.org/dc/terms/LCC	
Current definition	Library of Congress Classification
Current type of term	http://dublincore.org/usage/documents/principles/#encoding-scheme
Current "qualifies"	http://purl.org/dc/elements/1.1/subject
New definition	The set of concepts defined by Library of Congress Classification.
New type of term	http://purl.org/dc/dcam/VocabularyEncodingScheme

http://purl.org/dc/terms/UDC	
Current definition	Universal Decimal Classification
Current type of term	http://dublincore.org/usage/documents/principles/#encoding-scheme
Current "qualifies"	http://purl.org/dc/elements/1.1/subject

New definition	The set of concepts defined by Universal Decimal Classification.
New type of term	http://purl.org/dc/dcam/VocabularyEncodingScheme

http://purl.org/dc/terms/DCMIType	
Current definition	A list of types used to categorize the nature or genre of the content of the resource.
Current type of term	http://dublincore.org/usage/documents/principles/#encoding-scheme
Current "qualifies"	http://purl.org/dc/elements/1.1/type
New definition	A set of concepts used to categorize the nature or genre of the resource.
New type of term	http://purl.org/dc/dcam/VocabularyEncodingScheme

http://purl.org/dc/terms/IMT	
Current definition	The Internet media type of the resource.
Current type of term	http://dublincore.org/usage/documents/principles/#encoding-scheme
Current "qualifies"	http://purl.org/dc/elements/1.1/format
New definition	A set of concepts defined by IANA used to indicate the media type of the resource.
New type of term	http://purl.org/dc/dcam/VocabularyEncodingScheme

http://purl.org/dc/terms/ISO639-2	
Current definition	ISO 639-2: Codes for the representation of names of languages.
Current type of term	http://dublincore.org/usage/documents/principles/#encoding-scheme
Current "qualifies"	http://purl.org/dc/elements/1.1/language
New definition	The set of three-letter codes listed in ISO639-2 for the representation of names of languages.
New type of term	http://www.w3.org/2000/01/rdf-schema#Datatype

http://purl.org/dc/terms/RFC1766	
Current definition	Internet RFC 1766 'Tags for the identification of Language' specifies a two letter code taken from ISO 639, followed optionally by a two letter country code taken from ISO 3166.
Current type of term	http://dublincore.org/usage/documents/principles/#encoding-scheme
Current "qualifies"	http://purl.org/dc/elements/1.1/language
New definition	The set of tags, constructed according to RFC 1766, for the identification of languages.
New type of term	http://www.w3.org/2000/01/rdf-schema#Datatype

http://purl.org/dc/terms/URI	
Current definition	A URI Uniform Resource Identifier
Current see also	http://www.ietf.org/rfc/rfc2396.txt
Current type of term	http://dublincore.org/usage/documents/principles/#encoding-scheme
Current "qualifies"	http://purl.org/dc/elements/1.1/identifier
Current "qualifies"	http://purl.org/dc/elements/1.1/source
Current "qualifies"	http://purl.org/dc/elements/1.1/relation
New definition	The set of identifiers constructed according to the generic syntax for Uniform Resource Identifiers as defined by the IETF.
New see also	http://www.ietf.org/rfc/rfc3986.txt
New type of term	http://www.w3.org/2000/01/rdf-schema#Datatype

http://purl.org/dc/terms/Point	
Current definition	The DCMI Point identifies a point in space using its geographic coordinates.
Current type of term	http://dublincore.org/usage/documents/principles/#encoding-scheme
Current "qualifies"	http://purl.org/dc/terms/spatial
New definition	The set of points in space defined by their geographic coordinates according to the DCMI Point Encoding Scheme.

2007-08-25 Usage Board meeting, Singapore

New type of term	http://www.w3.org/2000/01/rdf-schema#Datatype

http://purl.org/dc/terms/ISO3166	
Current definition	ISO 3166 Codes for the representation of names of countries
Current type of term	http://dublincore.org/usage/documents/principles/#encoding-scheme
Current "qualifies"	http://purl.org/dc/terms/spatial
New definition	The set of codes listed in ISO 3166-1 for the representation of names of countries.
New type of term	http://www.w3.org/2000/01/rdf-schema#Datatype

http://purl.org/dc/terms/Box	
Current definition	The DCMI Box identifies a region of space using its geographic limits.
Current type of term	http://dublincore.org/usage/documents/principles/#encoding-scheme
Current "qualifies"	http://purl.org/dc/terms/spatial
New definition	The set of regions in space defined by their geographic coordinates according to the DCMI Box Encoding Scheme.
New type of term	http://www.w3.org/2000/01/rdf-schema#Datatype

http://purl.org/dc/terms/TGN	
Current definition	The Getty Thesaurus of Geographic Names
Current type of term	http://dublincore.org/usage/documents/principles/#encoding-scheme
Current "qualifies"	http://purl.org/dc/terms/spatial
New definition	The set of places defined by the Getty Thesaurus of Geographic Names.
New type of term	http://purl.org/dc/dcam/VocabularyEncodingScheme

http://purl.org/dc/terms/Period	
Current definition	A specification of the limits of a time interval.
Current type of term	http://dublincore.org/usage/documents/principles/#encoding-scheme
Current "qualifies"	http://purl.org/dc/elements/1.1/date
Current "qualifies"	http://purl.org/dc/terms/temporal
New definition	The set of time intervals defined by their limits according to the DCMI Period Encoding Scheme.
New type of term	http://www.w3.org/2000/01/rdf-schema#Datatype

http://purl.org/dc/terms/W3CDTF	
Current definition	W3C Encoding rules for dates and times - a profile based on ISO 8601
Current type of term	http://dublincore.org/usage/documents/principles/#encoding-scheme
Current "qualifies"	http://purl.org/dc/elements/1.1/date
Current "qualifies"	http://purl.org/dc/terms/temporal
New definition	The set of dates and times constructed according to the W3C Date and Time Formats Specification.
New type of term	http://www.w3.org/2000/01/rdf-schema#Datatype

http://purl.org/dc/terms/RFC3066	
Current definition	Internet RFC 3066 'Tags for the Identification of Languages' specifies a primary subtag which is a two-letter code taken from ISO 639 part 1 or a three-letter code taken from ISO 639 part 2, followed optionally by a two-letter country code taken from ISO 3166. When a language in ISO 639 has both a two-letter and three-letter code, use the two-letter code; when it has only a three-letter code, use the three-letter code. This RFC replaces RFC 1766.
Current type of term	http://dublincore.org/usage/documents/principles/#encoding-scheme
Current "qualifies"	http://purl.org/dc/elements/1.1/language
New definition	The set of tags, constructed according to RFC 3066, for the identification of languages.
New comment	RFC 3066 has been obsoleted by RFC 4646.
New type of term	http://www.w3.org/2000/01/rdf-schema#Datatype

2007-08-25 Usage Board meeting, Singapore

http://purl.org/dc/terms/NLM	
Current definition	National Library of Medicine Classification
Current type of term	http://dublincore.org/usage/documents/principles/#encoding-scheme
Current "qualifies"	http://purl.org/dc/elements/1.1/subject
New definition	The set of concepts defined by the National Library of Medicine Classification.
New type of term	http://purl.org/dc/dcam/VocabularyEncodingScheme

http://purl.org/dc/terms/RFC4646	
New label	RFC 4646
New definition	The set of tags, constructed according to RFC 4646, for the identification of languages.
New comment	RFC 4646 obsoletes RFC 3066.
New see also	http://www.ietf.org/rfc/rfc4646.txt
New type of term	http://www.w3.org/2000/01/rdf-schema#Datatype

http://purl.org/dc/dcam/memberOf	
Current label	Member Of
Current definition	A relationship between a resource and a vocabulary encoding scheme which indicates that the resource is a member of a set.
Current see also	http://dublincore.org/documents/2007/06/04/abstract-model/
Current type of term	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property

http://purl.org/dc/dcam/VocabularyEncodingScheme	
Current label	Vocabulary Encoding Scheme
Current definition	An enumerated set of resources.
Current see also	http://dublincore.org/documents/2007/06/04/abstract-model/
Current type of term	http://www.w3.org/2000/01/rdf-schema#Class

References

[DCAM]
Powell, Andy, Mikael Nilsson, Ambjörn Naeve, Pete Johnston and Thomas Baker. DCMI Abstract Model. DCMI Proposed Recommendation. February 2007.
<http://dublincore.org/documents/2007/02/05/abstract-model/>

[DC-RDF]

http://dublincore.org/documents/dc-rdf/

[DC-RDF-NOTES]

http://dublincore.org/documents/dc-rdf-notes/

[DCMES-CHANGES]

http://dublincore.org/usage/decisions/2006/2006-03.dcmes-changes.shtml

[DCTERMS]

http://dublincore.org/documents/dcmi-terms/

[DCTERMS-HISTORY]

2007-08-25	Usage Board meeting, Singapore

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

[DCQ-RDF]

http://dublincore.org/documents/2002/05/15/dcq-rdf-xml/

[DOMAINS]

http://dublincore.org/documents/2007/07/02/domain-range/

[NAMESPACE]

http://dublincore.org/documents/dcmi-namespace/

[NISOZ39-85]

http://www.niso.org/standards/resources/Z39-85-2007.pdf

[QUALIFIERS] http://dublincore.org/documents/2000/07/11/dcmes-qualifiers/

Metadata associated with this resource: http://dublincore.org/usage/public-comment/2007/07/dcterms-changes/index.shtml.rdf

Outline of http://dublincore.org/documents/dcmi-terms/ - a proposal

Identifier: http://dublincore.org/usageboardwiki/DcmiTermsOutline

2007-06-28 13:17

1. Introduction

2. Properties 2.1 The DCMES

    15 properties with /elements/1.1/ URIs

2.2 Other DCMI Properties

    Other properties with /terms/ URIs (including the new properties)

3. Vocabulary Encoding Schemes

4. Syntax Encoding Schemes

5. Classes 5.1 The DCMI Type Vocabulary

    The classes which are dcam:memberOf the dcterms:DCMIType VES

5.2 Other DCMI Classes (except for /dcam/ terms)

    New classes with /terms/ URIs

6. Terms related to the DCMI Abstract Model (/dcam/)

    dcam:memberOf and dcam:VocabularyEncodingScheme

Title: DCMI Usage Board Review of Application Profiles
Identifier: http://stage.dublincore.org/usage/meetings/2007/08/singapore/.html/profile-review.html
Source: e:/work/dcub/singapore/profile-review.txt
Created: 2007-06-27

The key documents for Usage Board review are:

-- Profile review criteria
http://dublincore.org/usageboardwiki/ProfileReviewCriteria

-- Term decision tree
http://dublincore.org/architecturewiki/TermDecisionTree

The latest work on "Description Set Profiles", which should be taken into account in revision the above:

-- Description Set Profile
http://dublincore.org/architecturewiki/DescriptionSetProfile

-- Introductory comments by Mikael
http://www.jiscmail.ac.uk/cgi-bin/webadmin?A2=ind0707&L=dc-architecture&P=1125

-- Eprints, converted into the Wiki format
http://knowware.nada.kth.se/DCWiki/EprintsApplicationProfile

Guidelines for application profiles, developed by the CEN workshop in 2004, which will need to be updated in light of the DSP model:

-- Dublin Core Application Profile Guidelines
http://dublincore.org/usage/documents/2005/09/03/profile-guidelines/

Process documents

-- Procedure for approval of proposals by DCMI
http://dublincore.org/documents/approval/ (updated August 2007)

-- DCMI Usage Board Administrative Process
http://dublincore.org/usage/documents/process/

Other relevant links:

-- DCMI Abstract Model
http://dublincore.org/documents/abstract-model/
-- Mixing and Matching FAQ ("why can't I just re-use my XML element", Andy, 2005)
http://www.ukoln.ac.uk/metadata/dcmi/mixing-matching-faq/
-- DCMI Policy on Naming Terms (uppercase for classes, lowercase for properties, etc, Tom and Stu, 2004)
http://dublincore.org/documents/naming-policy/
-- XML, RDF, and DCAPs (differences between an XML element and an RDF property, Pete, 2005)
http://www.ukoln.ac.uk/metadata/dcmi/dc-elem-prop/
-- Element Refinement in Dublin Core Metadata (discussion by Pete of semantic refinement, 2005)
http://dublincore.org/documents/dc-elem-refine/
-- Guidelines for Assigning Identifiers to Metadata Terms (Andy suggestions re: purl.org, info, xmlns.com, myproject.org, 2004)
http://www.ukoln.ac.uk/metadata/dcmi/term-identifier-guidelines/
-- Guidelines for using resource identifiers in Dublin Core metadata (a more recent version of the 2004 guidelines?)
http://dublincore.org/architecturewiki/ResourceIdentifierGuidelines
-- DC-TEXT
http://www.ukoln.ac.uk/metadata/dcmi/dc-text/
-- Guidelines for machine-processable representation of Dublin Core Application Profiles (CEN-workshop paper on representing application profiles in RDF, 2004)
ftp://ftp.cenorm.be/public/ws-mmi-dc/mmidc144.pdf

-- Legacy (2003) Usage Board definition of "application profile"
http://dublincore.org/usage/documents/profiles/index.shtml

ACTION 2007-06-08: Joe to revise
http://dublincore.org/usageboardwiki/ProfileReviewCriteria
in light of:
http://dublincore.org/usageboardwiki/CollectionsProfileReviewNotes

ACTION 2007-03-17: Stuart and Joe revise Term Decision Tree:
http://dublincore.org/architecturewiki/TermDecisionTree.
(Note: the difference is basically String vs. Thing.)

ACTION 2007-03-17: Joe to draft a document discussing
issues related to principles and purpose of UB decision-making.
(The context was the decision to define ISO639-2 as a set of codes.)
Joe will work with Stuart on this (2007-06-08).

ACTION 2007-03-17: Stuart and Joe to write
a one-page explanation differentiating
VES and SES, vet with Pete Johnston. See:
http://dublincore.org/usage/meetings/2007/03/barcelona/Encoding-schemes.txt.
Agreed: We need a deeper level of description and
differentiation between VES and SES, including definitions.
If you have a something already, how do you tell if it is
VES or SES. If an Encoding Scheme tells you what a value
string it it's a SES. If Encoding Scheme defines a class
of values, then it is a VES (e.g., concepts). For example,
if you develop a list of educational levels, and if you
define a list of strings, then you're defining an SES.
If you define a set of concepts and assign URIs to them (as
best practice), then you're defining a VES. Best practice
in this scenario is to define a set of concepts with URIs
rather than a set of strings. Agreed that DC-Education
is a great test-bed for these concepts. SES is a datatype
in RDF. VES is like conceptScheme in SKOS, only not limited
to concepts. For discussion: VES is a set of concepts that,
once in metadata, allows editors to handle assertion by
adding things to it. SES is a set of strings.

Usage Board review of application profiles: criteria and procedures

About this draft

This is work in progress by the Usage Board. Please do not cite or quote.

What are the boundaries of what constitutes Usage Board review? Need to distinguish between things that affect conformance and usage guidelines that may conform but with which the Usage Board may disagree. Is there a difference between semantic conformance and modeling conformance?

End product of review should be self-contained, including short description of a profile. Questions below should make reviewer assignments explicit enough to elicit such descriptions.

Usage Board Application Profile Review Guidelines (For Review: Barcelona)

General

- The assignment of *conforming* status by DCMI to an application profile indicates that at the time of submission for DCMI review: (1) the profile's usage of terms conforms to the DCMI Abstract Model; (2) the profile, taken as a whole, is internally consistent; and (3) the profile is sufficiently documented to serve the needs of the community of interest.
    - General guidelines for adequate documentation are set out below.
    - See also the *DCMI AP Application Profile*. (outline at Architecture Wiki: Outline of an Application Profile for an Application Profile)
- DCMI draws a jurisdictional distinction between:
    - Matters affecting a profile's conformance to the DCMI Abstract Model--matters upon which a Usage Board judgment of *conforming* status depends; and
    - Matters of conformant usage with which the Usage Board disagrees--matters upon which a Usage Board may only advise and recommend.
- While an application profile may be judged *conforming* by the DCMI, the assignment of *conforming* status does ***not*** mean that DCMI considers the conforming profile to be the only one that is useful for a particular community.

PROFILE PURPOSE AND SCOPE	
Question	**Consideration**
Are the purpose and scope of the AP clearly stated?	The documentation must define the goals of the profile in terms of the community of interest as well as the profile's purpose in terms of the resources to be described and the functionality it intends to support.
	The documentation should describe the context in which the application profile is used or is likely to be used.
	The documentation should identify the organizations or individuals involved in the profile's development as well as any arrangements, policies, or intentions regarding the future development and maintenance of the profile.
FUNCTIONAL REQUIREMENTS	
Question	**Consideration**

Are the functional requirements of the AP stated, and does the AP conform to the stated functional requirements?	The documentation must define the functional requirements of the profile. These requirements should be framed in terms of general functions such as (but not limited to) *discovery*, *identification*, and *selection* as well as in a more detailed enumeration of specific functionality enabled by the profile under each of the more general functions.
	The documentation must demonstrate that the application profile conforms to the stated functional requirements.
APPLICATION DATA MODEL	
Question	**Consideration**
Does the AP provide a coherent data model?	The Application profile must provide a data model that describes the profile's entities and the relationships among those entities. The data model may be illustrated in a graphical form (e.g., as one or more UML class diagrams) or set out in text.
	The application profile may be based on an externally expressed data model. In such a case, the application profile must clearly identify: (1) the external data model used; and (2) any points of divergence of the profile from that external model. Additional information deemed necessary to clarify the relationship between the profile and the external model should be provided.
DOCUMENTATION OF TERMS	
Question	**Consideration**
Are the terms used in the profile well described?	The elements used to describe the terms in the AP should conform to the CEN Guidelines in substance and labeling.
	The AP should use all appropriate descriptive elements to identify a term's *definitional attributes*, *identifying attributes*, *relational attributes*, and *constraints*.
Are *constraints* used consistently across the AP terms?	The AP should use *obligation*, *condition*, *data type*, and *occurrence* in a manner consistent with the functional requirements of the AP.
Do the recommended encoding schemes exist?	The recommended encoding schemes should exist and be declared in an existing namespace prior to Usage Board review.
CONFORMANCE OF INDIVIDUAL TERMS	
Question	**Consideration**
Does the term used in the AP conform to the *DCMI Abstract Model*?	Each term used in the AP should conform to the *DCMI Abstract Model*. Conformance should be confirmed by means of the DCMI Term Decision Tree.
Does the term usage in the AP represent a *refinement* and not a *re-definition* of the term used?	Terms used in an AP should refine and not re-define the semantics of the term used.
Are the decisions in the AP to declare a new term as opposed to refining an existing term sensible?	In creating an AP, developers are faced with the decision whether to refine an existing term through narrowed usage ***or*** to declare a new term that refines the original term. Where the AP-specific term usage solely restraints the term's value space, preference should be given to refining the original term through narrowed usage. Where the AP-specific term usage narrows the range of resources to which the term applies, the decision to create a new refining term or to use the original term restrained through a usage statement should be made based on the best interest of the community served.

Are the AP-specific encoding schemes appropriate?	{SAS NOTE: I am not sure what we mean by "appropriate" or how we operationalize it.}
Are the terms in the AP-specific encoding schemes adequately defined, sensible and conformant?	{SAS NOTE: I am not sure what "conformant" means in this context or how to operationalize it.}

Previous Draft and Discussions

- General
  - "Does the AP meet the community's needs?" *I think we decided this is the wrong question...*
- Documentation - introductory material
  - Purpose and scope
    - Are the purpose and scope of the AP clearly stated?
    - What is the stated goal? Cite and paraphrase.
  - Functional requirements
    - Are the functional requirements for the AP stated, and does the AP conform to the stated functional requirements?
    - *This needs to be expanded!*
  - Application Model
    - Does the data model make sense?
    - *This needs to be expanded!*
    - When reference is made to an externally expressed model, how much additional material do we expect?
- Documentation on terms
  - Are the terms well described - what descriptive elements are present?
  - How sensible are the labels for the descriptive elements?
  - Are the obligations consistent across the properties?
  - Do the recommended encoding schemes exist?
- Conformance of individual terms
  - Use the term decision tree, http://dublincore.org/architecturewiki/TermDecisionTree:
    - Check that each term conforms to the Abstract Model
    - Are any AP-specific encoding schemes appropriate?
    - Are the terms in the encoding scheme defined adequately, are the terms sensible, do they conform?
    - Making sure that "Usage in this DCAP" is *not* a case or re-definition.
    - We used to discourage people from creating new properties; now maybe encourage?

Issues arising from the UB assessment process, Manzanillo:

- The CEN guidelines need to be revised to follow what the UB sees as best practice in reviewing APs.
- When is it OK to use a general property but narrow its usage v when is it appropriate to define a new property?
- What is the convention in application profiles for displaying what comes from an external source and what is intrinsic to the specific application profile?
- We need to describe better how functional requirements fit into the application profile description, and where that relates to the application model.
- We should make it clear that we are not aiming to anoint particular application profiles as the only ones useful for a particular community.
- UB review needs to recognize that there may be a relationship between an externally expressed model and the AP, and the AP documentation might need to have some additional material to assist in relating the model to the AP

where dependencies exist.

- Question about whether we are explicitly endorsing subproperty assertions when we say that usage in an AP is conforming. If so we should say that up front.
- Issue of redefinition (or not) should be part of DCAP guidelines.
- Usage guidelines have to leave unchanged or narrow the semantics. And if you are narrowing, then why not define new property? We have been encouraging people to re-use. Rather, we could take the view of encouraging creation of new properties.

Application model (is it enough to support functional requirements?) Need one person looking at both together: Does it, on its face, meeting the requirements as stated? "This is a conforming way to say it in this particular context". Should be enough information in a Profile.

In the spirit of having stand-alone document. Usage guidelines - content rules documented here. Stand-alone documents define what the content standard is, if there is one.

Process: http://dublincore.org/usage/meetings/2006/09/manzanillo/profile-cdap/2006-02-13.process.txt

Three essential criteria are:

- 1. conformance to the abstract model
- 2. internal consistency
- 3. relationship of terms in the application profile to existing DC terms.

Review of Application Profiles (rough draft)

The mission of the Usage Board is to ensure an orderly evolution of the metadata terms maintained by the Dublin Core Metadata Initiative. The Usage Board evaluates proposals for new terms (or changes to existing terms) in light of grammatical principle, semantic clarity, usefulness, and overlap with existing terms. To proposals that are accepted, it assigns a specific status. The Usage Board also evaluates constructs that use DCMI terms, such as Application Profiles.

In order to do this the Usage Board must review proposals. Below is a set of guidelines for reviewing application profiles. There are four areas of evaluation and six criteria that can be applied to each area.

Four areas of evaluation: Application Functional Requirements, Application Domain Model, Description Set Profile, and Application Data Format

Six criteria are: Conform to the DCMI Abstract Model, Designed in non-conflict with grammatical principles (now DCMI description set profile), Internally consistent, Presented with semantic clarity, Useful to the community it serves, Does not introduce terms or other constructs that overlap with existing ones

All of these areas must be well documented.

Areas of evaluation:

Application Functional Requirements

- Question: Are the functional requirements of the profile described, and does the profile meet these functional requirements?
- Comments: The documentation should define the functional requirements of the profile.

These requirements should be formulated with regard to functions such as "finding", "identifying", and "selecting". Specific functions can be discussed in more detail if necessary. A good example: [1]. The documentation should show that the Description Set Profile meets the functional requirements.

- Question: Are the purpose and scope of a profile described clearly?
- Comments: The documentation packet should define the purpose with regard to:
    - the target group that will use the profile;
    - the resources that will be described with the profile, and
    - the functions that will be supported.
- The documentation packet should describe the context in which the application profile will be used (or can be used).
- The documentation should identify the organizations and individuals who participated in the development of the profile, along with any agreements, guidelines, or intentions regarding the future development and maintenance of the profile.

Application Domain Model

- Question: Are the entities in the world and the relationships among them that are to be described by the Application Profile described clearly; and are those entities and their relationships consistent with the Application Functional Requirements?

- Comment: An Application profile MUST provide an Application Domain Model, if only a simple one, which describes the entities and relationships amongst entities. The data model can be depicted in graphic form (e.g., as an UML class diagram) or in text.
- Comment: If the application domain model is based on a Standard Domain Model (e.g., FRBR), the Standard Domain Model must be identified and used clearly, consistently, and not in a conflicting manner? If the Application Domain Model deviates from the Standard Domain Model, the deviations must be well documented.

Description Set Profile

- Question: Does the Description Set Profile detail how to create a set of one or more descriptions, each of which describes a single [entity?} resource as set out in the Application Domain Model? [Redundant? And is this description set faithful to the Functional Requirements and Domain Model?]
- Question: Are newly declared terms documented?
- Discussion: The documentation should clearly define the terms.
- Comments:
- Questions: are vocabularies used in this AP clearly documented?
- Comments:
- Question: Are there user guidelines (optional)

Application Data Format

- Question: Does the application profile clearly demonstrate what syntax encoding is to be used?
- Comments:

Criteria for evaluating the four areas

Conform to DCMI Abstract Model

- Follows conventions of terminology
- Builds concepts of this model into the AP and its proposed use

Designed in non-conflict with Grammatical Principles

- One-to-one?, Dumb-down?
- Does the term usage in the AP represent a refinement and not a re-definition of the term used?

Terms used in an AP should refine and not re-define the semantics of the term used.

- Are the decisions in the AP to declare a new term as opposed to refining an existing term sensible? In creating an AP, developers are faced with the decision whether to refine an existing term through narrowed usage or to declare a new term that refines the original term. Where the AP-specific term usage solely restraints the term's value space, preference should be given to refining the original term through narrowed usage. Where the AP-specific term usage narrows the range of resources to which the term applies, the decision to create a new refining term or to use the original term restrained through a usage statement should be made based on the best interest of the community served.

- Are the AP-specific encoding schemes appropriate? {SAS NOTE: I am not sure what we mean by "appropriate" or how we operationalize it.}
- Are the terms in the AP-specific encoding schemes adequately defined, sensible and conformant? {SAS NOTE: I am not sure what "conformant" means in this context or how to operationalize it.} [2]

Internal Consistency (is the Application Profile internally consistent?)

- Does not contradict itself
- Does not leave concepts or constructs ambiguous
- Consistently cites outside sources where necessary, using their terminology if not purposefully and clearly changing it in the AP

Presented with semantic clarity

- Terms, concepts, constructs, and citations are presented clearly and meaningfully.
- Are the terms used in the profile well described? The elements used to describe the terms in the AP should conform to the CEN Guidelines in substance and labeling. The AP should use all appropriate descriptive elements to identify a term's definitional attributes, identifying attributes, relational attributes, and constraints.
- Are constraints used consistently across the AP terms? The AP should use obligation, condition, data type, and occurrence in a manner consistent with the functional requirements of the AP.

Useful to the community it serves

- Well documented record of community and how this profile will be useful to it. Demonstrated feedback and vetting.

Does not overlap with terms or constructs approved by the DCMI Usage Board

[1] http://www.ukoln.ac.uk/repositories/digirep/index/Functional_Requirements [2] http://dublincore.org/usageboardwiki/ProfileReviewCriteria

2007-08-25 Usage Board meeting, Singapore

DCMI-compliant 'term' decision tree

AndyPowell
Eduserv Foundation
December 2005

This document is currently under development. It is being worked on by the Self:DC RDF Taskforce. Comments should be sent to the dc-rdf-taskforce@jiscmail.ac.uk mailing list.

Introduction

This decision tree can be used to see if something is a DCMI-compliant element, element refinement or encoding scheme, where "DCMI-compliant" means conformant with the DCMI Abstract Model and therefore suitable for use in DC metadata descriptions.

Note that in the following text, the italicised terms are defined in the terminology section below.

Decision tree

1. Has the thing been explicitly declared as a DCMI *element* (i.e. as an RDF *property*)?
   The declaration should take the form of a human-readable statement, e.g.

   ```
   X is a DCMI 'element'
   ```

   or

   ```
   X is an RDF property.
   ```

   and a machine-readable RDFS declaration

   ```
   <rdf:Property rdf:about="http://example.org/term/X">
     ...
   </rdf:Property>
   ```

   If 'yes', go to question 2. Otherwise, go to question 3.
2. Have the expected *values* of the *element* been assigned *value URIs* or can they be represented using simple *value strings* (plain text strings)?
   If 'yes', go to question 9. Otherwise, go to question 3.
3. Has the thing been explicitly declared as a DCMI *element refinement* (i.e. as an RDF *property*)?
   The declaration should take the form of a human-readable statement, e.g.

   ```
   X is a DCMI 'element refinement'.
   ```

   or

```
X is an RDF property.
```

and a machine-readable RDFS declaration

```
<rdf:Property rdf:about="http://example.org/term/X">
  ...
</rdf:Property>
```

If 'yes', go to question 4. Otherwise, go to question 5.

4. Have the expected *values* of the *element refinement* been assigned *value URIs* or can they be represented using simple *value strings* (plain text strings)?

   If 'yes', go to question 9. Otherwise, go to question 5.

5. Has the thing been explicitly declared as a DCMI *syntax encoding scheme*?

   The declaration should take the form of a human-readable statement, e.g.

```
X is a DCMI 'syntax encoding scheme'.
```

or

```
X is an RDF datatype.
```

and a machine-readable RDFS declaration

```
<rdfs:Datatype rdf:about="http://example.org/term/X">
  ...
</rdfs:Datatype>
```

If 'yes', go to question 6. Otherwise, go to question 7.

6. Are all the valid constructs according to the *syntax encoding scheme* simple *value strings* (plain text strings)?

   If 'yes', go to question 9. Otherwise, go to question 7.

7. Has the thing been explicitly declared as a DCMI *vocabulary encoding scheme*?

The declaration should take the form of a human-readable statement, e.g.

```
X is a DCMI 'vocabulary encoding scheme'.
```

or

```
X is an RDF class.
```

and a machine-readable RDFS declaration

```
<rdfs:Class rdf:about="http://example.org/term/X">
  ...
</rdfs:Class>
```

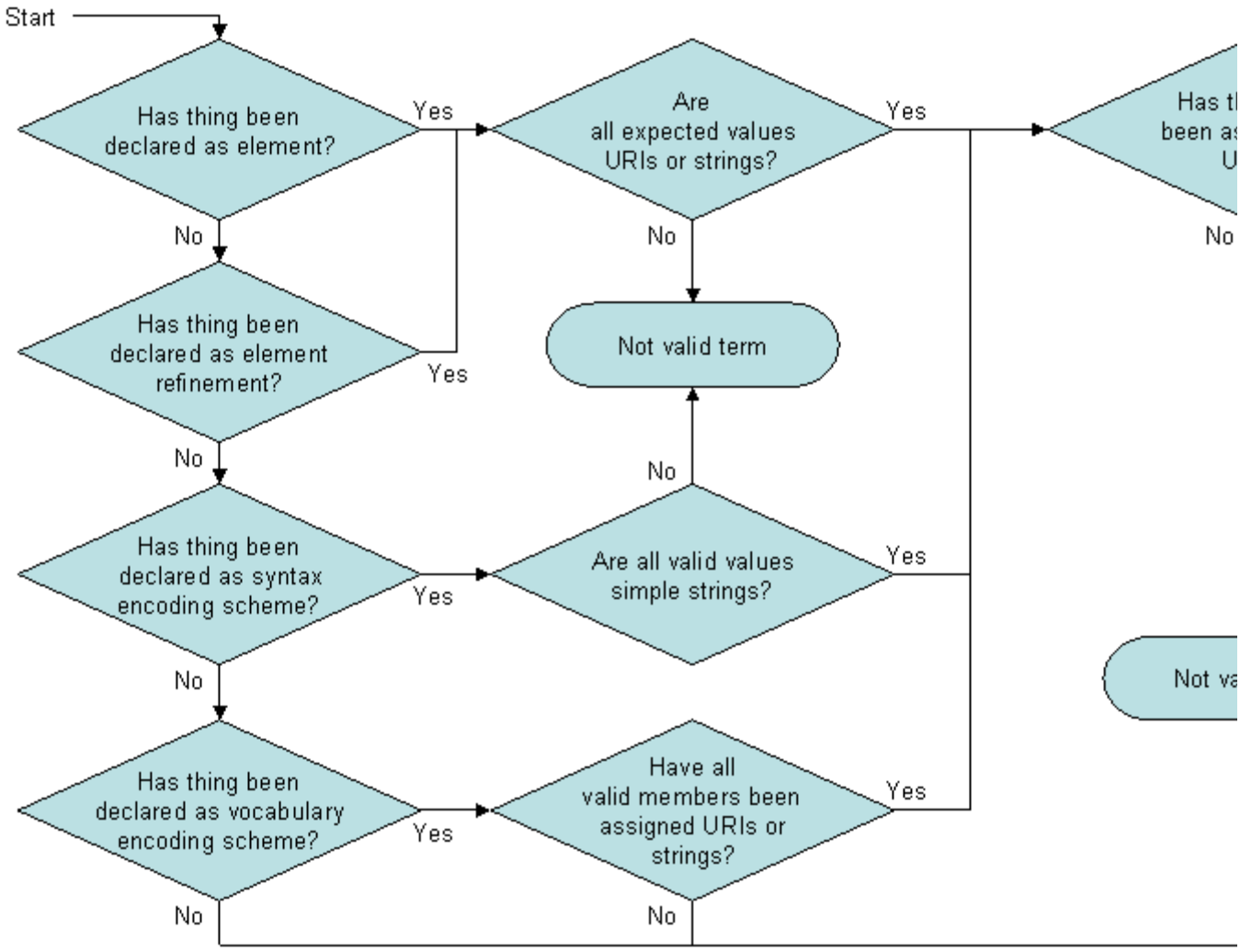
If 'yes', go to question 9. Otherwise, the thing is not a valid DCMI *element*, *element refinement* or *encoding scheme*.

8. Have all the valid members of the *vocabulary encoding scheme resources* been assigned *value URIs* or can they be represented using simple *value strings* (plain text strings)?

   If 'yes', go to question 9. Otherwise, the thing is not a valid DCMI *element*, *element refinement* or *encoding scheme*.

9. Has the thing been assigned a URI (a *property URI* or an *encoding scheme URI*)?

   For example: http://example.org/term/X . Dereferencing the URI should result in an HTTP 303 redirect to HTML (text/html) and RDF (application/rdf+xml) representations of the *term*. HTTP content negotiation should be used to select one or other representation. **If 'yes', the thing is a valid DCMI *element, element refinement* or *encoding scheme*.** Otherwise, the thing is not a valid DCMI *element*, *element refinement* or *encoding scheme*.



DC-compliant term decision tree

Notes

Best practice for serving representations of metadata terms using HTTP is still emerging. DCMI suggests the approach recommended above. Further, DCMI suggests serving an RDF declaration for each term that includes enough contextual information to interpret the term in the context of related terms. For example, it may be sensible to serve an RDF declaration for all the terms in a given namespace as the representation of each of the individual terms within that namespace.

New *terms* that are proposed during the creation of an application profile may be temporarily assigned *term URIs* using the `example.org` domain name, pending the assignment of a proper *term URI* by the DCMI Usage Board. In this case, the criteria concerning dereferencing the URI to obtain a human-readbale or

machine-readable representation of the *term* do not apply.

2007-08-25 Usage Board meeting, Singapore

The diagram above is also available as a MS-Powerpoint file.

Terminology

''class''

A group containing members that have attributes, behaviours, relationships or semantics in common; a kind of category.

''class URI''

A URI that identifies a *class*.

''element''

A *property* of a *resource*.

''element refinement''

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.

''encoding scheme''

A *vocabulary encoding scheme* or a *syntax encoding scheme*.

''encoding scheme URI''

A *vocabulary encoding scheme URI* or a *syntax encoding scheme URI*.

''property''

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

''property URI''

A URI that identifies a single *property*.

''syntax encoding scheme''

An indication that the *value string*
is formatted in accordance with a formal notation, such as "2000-01-01" as the standard expression of a date.

''syntax encoding scheme URI''

A URI that identifies a *syntax encoding scheme*.

''term''

A *property* (i.e. *element* or *element refinement*), *vocabulary encoding scheme*, *syntax encoding scheme* or concept taken from a controlled vocabulary (concept space).

''term URI''

A URI that identifies a *term*.

''vocabulary encoding scheme''

A *class* that indicates that the *value* of a *property* is taken from a controlled vocabulary (or concept-space), such as the Library of Congress Subject Headings.

''vocabulary encoding scheme URI''

A URI that identifies a *vocabulary encoding scheme*.

SES and VES

Joe Tennis and Stuart Sutton

According to the DCAM:

- A vocabulary encoding scheme is an enumerated set of resources.
- A syntax encoding schemes is a set of strings and an associated set of rules that describe a mapping between that set of strings and a set of resources. The mapping rules may define how the string is structured (for example DCMI Box) or they may simply enumerate all the strings and the corresponding resources (for example ISO 3166).

Problem 1: Neither of these definitions is sufficient to identify a controlled vocabularies used for indexing resources. Why?

A controlled vocabulary is a list of terms that have been enumerated explicitly. This list is controlled by and is available from a controlled vocabulary registration authority. All terms in a controlled vocabulary must have an unambiguous, non-redundant definition.

NOTE: This is a design goal that may not be true in practice; it depends on how strict the controlled vocabulary registration authority is regarding registration of terms into a controlled vocabulary.

At a minimum, the following two rules must be enforced:

- 1. If the same term is commonly used to mean different concepts, then its name is explicitly qualified to resolve this ambiguity. NOTE: This rule does not apply to synonym rings.
- 2. If multiple terms are used to mean the same thing, one of the terms is identified as the preferred term in the controlled vocabulary and the other terms are listed as synonyms or aliases. [1]

As can be seen from this standard definition, we are not talking about resources, we are talking about terms and concepts these are not...:

Problem 2: These definitions are not clearly mutually exclusive. Why?

If we were to implement a controlled vocabulary in a metadata record, we not only would pick from a list (the implication of a VES), but we would need to put a set of strings in the record that was a mapping between that string and a resource (or resources) that had mapping rules.

The Proposed Amelioration for Problem 2: URIs are not Strings

The amelioration proposed for the problem stated above is that if the value is a URI it is a VES and if it is a string (not a URI) then it is a SES. However, this is a problem with modeling representation. The string and the URI are both stand-ins for the thing itself, and both have some form a representation at work (mapping from string to thing). The URI is a not an essential difference, and is one that may be important for syntax processing but is not true to the Standard or Application models of working with controlled vocabularies .

Proposed action: Add a controlled vocabularies encoding scheme to the Abstract Model that accounts for the definition in [1]. This CVES which would be a subset of VES.

[1] http://www.niso.org/standards/resources/Z39-19-2005.pdf

Title: DCMI Usage Board Review of Collections Application Profile
Date: 2007-07-20
Description: This document is the result of a March 2007 review by the DCMI Usage Board of the "Dublin Core Collections Application Profile".

About the profile

The Collections Application Profile [1] was developed in order to provide a method for describing "collections" -- both collections of resources, and catalogues of collections of resources (in essence a specialised form of "collection") -- which conforms to the DCMI Abstract Model [2]. The Collections Application Profile defines collection as "any aggregation of physical and/or digital resources". Examples of collections include aggregations of natural (i.e. physical) objects, created objects, "born-digital" items, and digital surrogates of physical items. The Collections Application Profile is not intended to describe every possible characteristic of resource collections, as its primary purpose is discovery.

The simple descriptions provided for in the Collections Application Profile are suitable for describing many different types of collections. Descriptions using this Application Profile are primarily designed to support:

* discovery of collections;
* identification of collections;
* selection of collections;
* identification of collection locations;
* identification of services providing access to collections.

The domain model used in the Collections Application Profile is based on "An analytical model of collections and their catalogues" by Michael Heaney [9]. This analytical model focuses on Collections. Collections hold Items, are described by Catalogues or Indexes, are located in Locations and accessed via Services, and are owned or collected by Owners who provide Services.

Descriptions of Collections specify the nature of a collection, its name, size, language, topic, intended audience, and scope, as well as the methods and policies by which items are added to the collection. In addition, descriptions indicate the Collector and Owner of the collection, point to the location where the collection is held and services which provide access, and note any related collections, catalogues, or publications.

In order to do this, the Collections Application Profile uses DCMI terms along with more specialized terms created for use in the profile [3]. Controlled vocabularies of values were created for use in this profile: the Dublin Core Collection Description Type Vocabulary [4], an Accrual Method Vocabulary [5], an Accrual Policy Vocabulary [6], and a Frequency Vocabulary [7]. A summary of terms and constraints used in the profile was compiled into a tabular list [8].

Building on prior work of the UKOLN Collection Description Focus [11] on the Resource Support Libraries Programme [12] (see also [13]), the DCMI Collection Description Working Group began in 2002 [10]. The initial development of the Collections Application Profile was led by Pete Johnston,

chair of the Collections Description Working Group [15]. This working group was characterized by an interested and involved community which provided lengthy and valuable comments on various drafts of the Application Profile, showing a high-level of buy-in to the Application Profile by the community.

The DCMI Usage Board undertook an informal review of the draft profile in September 2005. The Application Profile was then formally submitted to the Usage Board for a review in mid-2006, resulting in further feedback to the Working Group. When Pete stood down as chair of the Group in 2006, work on the Application profile was taken over by Sarah Shreeves and Muriel Foulonneau. Sarah Shreeves and Muriel Foulonneau submitted a revised version of the Collections Application Profile to the Usage Board in March 2007 and the Profile was reviewed by the Usage Board at its March 2007 meeting. Although some minor issues were flagged (see below), the Usage Board approved the profile as a conforming Application Profile.

Review criteria and result

The Usage Board determined that the Collections Application Profile "conforms", which in this context was taken to mean:

-- the profile's usage of terms conforms to the DCMI Abstract Model;

-- profile, taken as a whole, is internally consistent; and

-- the profile is sufficiently documented to serve the needs of the community of interest.

Other comments (not related to conformance per se)

-- The Usage Board has long recognized that, in practice, the dc:identifier property is used both for identification and for location. The property cld:isLocatedAt, a sub-property of dc:relation, refers to physical locations. Some reviewers felt that expressing digital locations with dc:identifier and physical locations with a separate property could prove to be problematic, especially for mixed physical and digital collections.

-- Some reviewers questioned whether it is helpful to define cld:isLocatedAt as a subproperty of dc:relation (hence subject to being resolved to dc:relation during dumb down).

-- In the Purpose and Scope section, the text states: "Those resources may be of any type, so examples might include [...] catalogues of such collections (as aggregations of metadata records)." This sentence implies a generally understood library-world meaning of "record". Some reviewers saw a potential for confusion with the different, more specific concept of "record" defined in the DCMI Abstract Model.

References

[1] http://dublincore.org/groups/collections/collection-application-profile/2007-03-09/
[2] http://dublincore.org/documents/2007/04/02/abstract-model/
[3] http://dublincore.org/groups/collections/collection-terms/2007-03-09/
[4] http://dublincore.org/groups/collections/colldesc-type/2007-03-09/
[5] http://dublincore.org/groups/collections/accrual-method/2007-03-09/
[6] http://dublincore.org/groups/collections/accrual-policy/2007-03-09/

[7] http://dublincore.org/groups/collections/frequency/2007-03-09/
[8] http://dublincore.org/groups/collections/collection-ap-summary/2007-03-09/
[9] http://www.ukoln.ac.uk/metadata/rslp/model/
[10] http://www.jiscmail.ac.uk/cgi-bin/webadmin?A2=ind0202&L=dc-collections&P=60
[11] http://www.ukoln.ac.uk/cd-focus/
[12] http://www.ukoln.ac.uk/metadata/rslp/
[13] http://dlib.ukoln.ac.uk/dlib/september00/powell/09powell.html
[14] http://dublincore.org/usage/meetings/2007/03/barcelona/Topic-cdap.txt
[15] http://dublincore.org/groups/collections/

Procedure for approval of proposals by DCMI

Date Issued:	2007-08-06
Identifier:	http://dublincore.org/documents/2007/08/06/approval/
Replaces:	http://dublincore.org/usage/documents/2004/12/20/approval/
Is Replaced By:	Not applicable
Latest version:	http://dublincore.org/documents/approval/
Status of document	This is a DCMI Process Document
Description of document:	This document lists the step-by-step process for the approval of five types of proposals: 1. Proposed changes to DCMI metadata terms 2. Proposals for DCMI Recommendations 3. Proposals for DCMI Recommended Resources 4. Proposals for Application Profiles as DCMI Recommended Resources 5. Proposals for DCMI Process Documents.

About proposals in general

There are five categories of proposals:

1. Proposed changes to DCMI metadata terms
2. Proposals for DCMI Recommendations
3. Proposals for DCMI Recommended Resources
4. Proposals for Application Profiles as DCMI Recommended Resources
5. Proposals for DCMI Process Documents

Proposals can come from the Directorate (e.g. as a result of an outsourced activity), Affiliates, Communities, Task Groups, other organizations external to DCMI, or any individual.

Any document that goes into the approval process needs to be submitted to the DCMI Managing Director using the Proposal Submission Form. Proposals may be submitted as a link to an external resource in plain text, HTML or PDF.

The Directorate acknowledges receipt and decides whether a document falls in one of the five categories, possibly in consultation with the Advisory Board. A first decision on whether DCMI will accept a proposal for consideration is communicated to the submitter no later than two months after submission with specification of the process and timeline foreseen.

1. Proposals related to DCMI metadata terms

Proposals related to existing DCMI metadata terms (including editorial comments or updating obsolete references) are evaluated by the DCMI Usage Board within four weeks from submission. If the advice from the Usage Board is negative, the Directorate rejects the proposal and informs the submitter with reasons for rejection. If the advice is positive and the proposed change is substantial, the Directorate prepares for public comment. Minor editorial changes will be made immediately without a public comment period.

Any comments from the UB are communicated to submitters who have the opportunity to make modifications based upon those comments. If major changes are made, the proposal needs to be re-submitted.

Public comment is open for a minimum of four weeks, announced by the Directorate on DC-GENERAL and with a news item on the DCMI Web site and published on the Public Comment page. If serious objections are expressed by the public, the Directorate decides what to do, in consultation with the Usage Board: (a) reject the proposal and inform the submitter with reasons, giving the option to re-submit with changes, or (b) ask submitters to make minor changes to the proposal.

If no serious objections are received, the Usage Board incorporates the proposal in the documentation of Dublin Core terms, after which the Directorate publishes an updated version of the DCMI Metadata Terms on the DCMI Web site and announces to DC-GENERAL and with a news item on the DCMI Web site.

2. Proposals for other DCMI Recommendations

Examples: DCMI Abstract Model, Namespace Policy

The DCMI Directorate assigns the status of Proposed Recommendation or (in cases where a Recommendation already exists) Proposed Revised Recommendation.

Proposals go through an initial review in the Advisory Board for at least four weeks. After negative review, the Directorate informs the submitter with reasons for rejection. Modified proposals can be re-submitted. If AB review is positive, the Directorate prepares for public comment. Any comments from the AB are communicated to submitters who get the opportunity to make modifications based upon those comments. If major changes are made, the proposal needs to be re-submitted.

Public comment is open for a minimum of four weeks, announced by the Directorate on DC-GENERAL and with a news item on the DCMI Web site and published on the Public Comment page. If serious objections are expressed by the public, the Directorate decides what to do, in consultation with the Advisory Board: (a) reject the proposal and inform the submitter with reasons, giving the option to re-submit with changes, or (b) ask submitters to make minor changes to the proposal.

After successful public comment, the Directorate publishes the Recommendation on the DCMI Web site and announces to DC-GENERAL and with a news item on the DCMI Web site.

3. Proposals for DCMI Recommended Resources

Examples: AskDCMI, Using Dublin Core

In general, evaluation and subsequent acceptance or rejection of proposed DCMI Recommended Resources are at the discretion of the Directorate, in consultation with the Advisory Board. The procedure for Application Profiles is different and is described in Section 4 below.

The decision on acceptance or rejection is taken by the Directorate and communicated to the submitter no later than six months after the submission of the proposal. Rejections are communicated to the submitter with reasons for rejection. The Directorate announces the acceptance of proposals to DC-GENERAL and with a news item on the DCMI Web site. The Recommended Resource is published on the DCMI Web site (possibly as a link to the resource, for example as is the case with the question-and-answer service AskDCMI).

2007-08-25 Usage Board meeting, Singapore

4. Proposals for Application Profiles as DCMI Recommended Resources

Application Profiles submitted for review are assigned to the Usage Board and evaluated according to published criteria. The Usage Board advises the Directorate on acceptance or rejection of such proposals.

Decision on acceptance or rejection is taken by the Directorate and communicated to the submitter no later than six months after the submission of the proposal. Rejections are communicated to the submitter with reasons for rejection. The Directorate announces the acceptance of proposals to DC-GENERAL and with a news item on the DCMI Web site. The Application Profile is published as a Recommended Resource on the DCMI Web site.

5. Proposals for DCMI Process Documents

Examples: DCMI By-laws, Procedure for approval of proposals by DCMI (this document)

Evaluation and subsequent acceptance or rejection of the proposal is at the discretion of the Directorate, in consultation with the Advisory Board.

Decisions on acceptance or rejection are taken by the Directorate and communicated to the submitter no later than two months after the submission of the proposal. Rejections are communicated to the submitter with reasons for rejection. The Directorate announces the acceptance of proposals to the DC-GENERAL and with a news item on the DCMI Web site. The process document is published as a DCMI Process Document on the DCMI Web site.

2007-08-25 Usage Board meeting, Singapore

DCMI Usage Board Administrative Processes

Creator:	Diane I. Hillmann
Creator:	Stuart A. Sutton
Contributor:	Thomas Baker
Date Issued:	2006-02-13
Identifier:	http://dublincore.org/usage/documents/2006/02/13/process/
Replaces:	http://dublincore.org/usage/documents/2003/02/07/process/
Is Replaced By:	Not Applicable
Latest version:	http://www.dublincore.org/usage/documents/process/
Status of document:	This is a DCMI Process Document.
Description of document:	This document describes the process by which the DCMI Usage Board reaches and documents decisions. The Usage Board (UB) acts in accordance with its charter under the DCMI Bylaws, Article II, section D. The descriptions of process in this document are intended to guide the UB in executing its responsibilities for ensuring "an orderly evolution of the metadata terms maintained by the Dublin Core Metadata Initiative". The process statements are amended from time to time to reflect the evolving role of the Usage Board. In case of discrepancies, the DCMI Bylaws control.

Index

- Usage Board Membership
- Meetings
- Status assigned by Usage Board decisions
- Process for moving proposals
- Registration of Application Profiles
- New terms in Application Profiles
- Endorsement of terms in other namespaces
- Revisions of existing terms

1. Usage Board Membership [from DCMI Bylaws].

1.1. Membership. The Usage Board will consist of at least seven and no more than eleven people (nine is ideal) appointed by the DCMI Directorate.

1.2. Responsibilities. The mission of the Usage Board is to ensure an orderly evolution of the metadata terms maintained by the Dublin Core Metadata Initiative. The Usage Board evaluates proposals for new terms (or changes to existing terms) in light of grammatical principle, semantic clarity, usefulness, and overlap with existing terms. To proposals that are accepted, it assigns a specific status. The Usage Board also evaluates constructs that use DCMI terms, such as Application Profiles.

1.3. Selection and Appointment Process. Members are selected based on the following criteria: knowledgeable concerning the development history and purpose of the DC element set and its relationship to the metadata world at large; related to a metadata community relevant to DCMI; willing and able to commit time and energy to the functions of the UB; able to communicate verbally and in writing in English well enough to prepare documents and discuss complex issues in a group setting; geographic and domain distribution of members is relevant but will not override other criteria. The DCMI Directorate will appoint the UB Chair from one of the membership. The DCMI Directorate can propose the appointment of non-voting Liaison members to the Usage Board. Liaison members may represent DCMI Affiliates or Sponsors, or other organizations that have a stake in the development of the Dublin Core semantic specifications.

1.4. Terms. Usage Board members are appointed for two-year, renewable terms. They may step down at their own discretion at any time.

1.5. Decision process. The Usage Board strives for consensus, justifying its decisions and interpretations in terms both of principle and of empirical practice. To be approved, a proposal needs more than 50% of assigned votes in favor and less than 25% of assigned votes against. Important decisions will be assigned a number for citation purposes and documented on the DCMI Web site. Decisions of the UB are forwarded to the DCMI Directorate for endorsement and approval.

1.6. Communication and documentation. For internal communication, the UB uses a closed mailing list. The messages are archived and are made publicly available. Meetings are held at least once a year. This meeting is scheduled during the annual DC general workshop/conference. Further meetings can be scheduled, preferably close to other conferences, so as to make attendance convenient for as many members as possible. Scheduling is done far enough in advance so that as many members as possible may be present.

1.7. Reporting. The chair of the Usage Board is responsible for the preparation of a report of meetings and conference calls and submission to the DCMI Directorate. Based on this report and after endorsement of Usage Board decisions, the Managing Director communicates the decisions to the DCMI community. Decisions on semantics are included in the reference documentation on the DCMI Web site.

2. Meetings

2.1. The calendar of UB meetings will be announced prominently on the UB DCMI homepage and announced on the DC-General mailing list.

2.2. Funding for regular UB members attendance at meetings should be supported as much as possible by DCMI. Funding for the attendance of Liaisons at UB meetings should be provided by their institutions.

2.3. The UB Chair maintains the agenda, which cites links to relevant supporting documentation, including JISCMAIL postings. All materials considered by means of the agenda are consolidated in a PDF briefing book and distributed electronically to UB members and Liaisons. At the conclusion of the meeting, the PDF briefing book becomes

the official record of matters considered at the meeting. Important decisions will be assigned a number for citation purposes and documented on the DCMI website.

2.4. Additional meetings of the UB through conference calls may be scheduled at the discretion of the UB Chair.

2.5. The UB chair is responsible for assigning shepherds to proposals. Agenda items shall include the name of the UB member responsible for shepherding the proposal through the UB process. Agendas shall be available at the Web page DCMI Usage Board Meetings several weeks before the meeting.

2.6. Members must attend at least one meeting in a given year to maintain membership in good standing. Members who miss two meetings in succession may be replaced by the DC Directorate.

2.7. Attendance at any UB meeting by other than the UB members and liaisons is by invitation. People interested in attending should request an invitation via the UB Chair or the Managing Director. Participation in discussion of proposals by any interested parties is encouraged.

3. Status assigned by Usage Board decisions

Conforming [URI http://dublincore.org/usage/documents/process/#conforming]. Elements, Element Refinements, and Application Profiles may be assigned a status of conforming. Elements and Element Refinements assigned a status of conforming are those for which an implementation community has a demonstrated need and which conform to the DCMI Abstract Model.

Recommended [URI http://dublincore.org/usage/documents/process/#recommended]. Elements, Element Refinements, and DCMI-maintained Vocabulary Terms (e.g., member terms of the DCMI Type Vocabulary) that conform to the DCMI Abstract Model and do not semantically overlap with other terms in DCMI namespaces (i.e., http://purl.org/dc/elements/1.1/ and http://purl.org/dc/terms/) may be assigned the status of Recommended.

Obsolete [URI http://dublincore.org/usage/documents/process/#obsolete]. For Elements and Element Refinements that have been superseded, deprecated, or rendered obsolete. Such terms will remain in the registry for use in interpreting legacy metadata.

Registered [URI http://dublincore.org/usage/documents/process/#registered]. Used for Vocabulary Encoding Schemes and language translations for which the DCMI provides information but not necessarily a specific recommendation.

Endorsed [URI http://dublincore.org/usage/documents/process/#endorsed]. A non-DCMI assertion may be assigned the status of Endorsed where: (1) the term is managed by a registration authority other than DCMI and the assertion is that the term is conforming to the DCMI Abstract Model; or (2) the term is managed by a registration authority other than DCMI and the assertion is that the term bears either a property or subproperty relationship to a DCMI term.

5. Process for moving proposals

5.1. Assignment of shepherd Each impending proposal and application profile shall be assigned a shepherd by the UB chair from among the UB membership at the earliest opportunity. Shepherds should have knowledge of the proposal issues or application profile domain.

5.2. Shepherd responsibilities include:

- Monitoring discussion on relevant lists (shepherds should be members of the relevant DC WG list during the time of consideration of a proposal and are encouraged to join in the discussion to ensure that all relevant issues are exposed during the discussion period).
- Summarizing the comment period discussion and points of contention of the proposal for the UB, either verbally at the meeting or in writing prior to the meeting (preferred). Serving as liaison to the relevant WG or community during the time the proposal is under discussion and after a decision has been made. Preparing a draft of UB official decision on the proposal for review and approval by the UB.
- In general providing advice and expertise to the Working Group or domain on good practices, the Abstract Model, and other issues affecting the process of developing a proposal or application profile. The shepherd should bring issues of concern to the attention of the UB when appropriate.

5.3. Preparing for public comment periods. Completed proposals are forwarded to DCMI Managing Director or UB Chair. Proposals are given preliminary review for completeness by the DCMI Managing Director and UB Chair. If complete and no revisions are needed, proposals are circulated to UB members and announced for public comment by the Managing Director. A period of two weeks will be allowed between the date of the decision on completeness and the public announcement of the proposal to provide time for preparation of the supporting materials for public dissemination. If incomplete or revisions are needed, proposals are returned to the originator, with a request for revision or additional information.

5.4. Announcing the public comment period. Before announcement of the public comment period, proposals must be moved to the DCMI Web site, given DCMI page headers and a status of 'Proposed'. Announcement of the public comment period shall be made on the DC-General mailing list by the head of UB. Announcements should include links to the proposal; links to other relevant information; deadlines for comments; email addresses to be used for submitting comments. (In general, comments regarding a proposal may be addressed to the relevant WG mailing list, the DC-General mailing list or privately to the shepherd.) Announcements may also include information about the UB meeting at which the proposal will be discussed, including place, time, and how to request an invitation to participate; name and contact information for the assigned shepherd. The announcement should ask specifically for communications supporting the proposal in order to gauge the level of community support.

5.5. Managing public comment periods. The comment period for proposals should be managed on the DC-General list. Comment periods must be at least one month in length and commence at least six weeks before the UB meeting at which action is to be taken. Public discussions of UB related issues during public comment periods should take place on DC-General or other working group mailing lists as specified in the announcement. The public discussion must start at least six weeks before the UB meeting at which the issues will be discussed.

5.6. Communication responsibility

Communication Responsibility Table

What	Where	Who	Comment
Proposal draft posted	WG list, DC-General	WG Chair	
Proposal added to UB agenda	UB Website, UB list	UB Chair	
Proposal announced for public comment	DC-General	DCMI Directorate	

Usage Board decision announced	DC-General	DCMI Directorate	

5.7. Voting on proposals. Voting shall be limited to scheduled meetings and publicly announced conference calls. Voting shall be limited to UB members present at the meeting or conference call and able to participate in the discussion. UB members who cannot be present may present their arguments for or against a proposal in writing prior to a meeting (this shall not constitute a vote). UB members who cannot be present may explore other options with the chair, if they cannot be present for an important vote. In all cases, a vote may not be cast by a member who is not present, either physically or virtually, for the relevant discussion.

5.9. Documenting Usage Board decisions. A document explaining the UB decision regarding a proposal will be written in a timely fashion by the shepherd and approved by the UB. The decision will include brief statements of recommendations being issued and detailed explanations of UB decisions not to issue recommendations. UB decisions will be in a form determined by the UB and numbered consecutively for the purpose of citation. UB decisions must be sufficiently documented so that the rationale for the decision is clear and useful in guiding the development of future proposals. This is particularly true where the decision rejects a proposal or recommends further action.

5.10. Announcing and publishing Usage Board decisions. Decisions are published on the Web page DCMI Usage Board decisions. New terms will be added to the official DCMI documentation by the UB Chair.

6. Proposals for registration of application profiles

6.1. Application Profiles subject to review. Application profiles emanating from DCMI Strategic Activities may be reviewed by the Usage Board. Metadata implementers (established projects, communities or research groups) may also request review, subject to approval by the UB Chair. *Point to information regarding DCMI Strategic Activities when available.*

6.2. Documentation of Application Profiles. 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. 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. It is recommended that application profiles be prepared using the guidelines "Dublin Core Application Profile Guidelines".

6.4. Contextual information about Application Profiles. The documentation for each Application Profile must provide -- or point to a short text that describes -- the context and purposes in which the application profile is used or is likely to be used; the organizations or individuals involved in its development and a capsule history thereof; and any arrangements, policies, or intentions regarding the future development and maintenance of the application profile.

6.5. Evaluation of terms in Application Profiles. 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. *Note: revisit this.*

6.6. Assignment of status "conforming". Application profiles which pass review will be assigned the status of conforming The status of conforming indicates a Usage Board assessment of the application profile as of the date of its submission for review. Changes to already conforming application profiles require further Usage Board review of the application profile in whole or in part according to the processes and criteria

outlined in sections 6.1 through 6.3.

6.7. Publication of Usage Board reviews of Application Profiles. For application profiles that "pass" review, the Usage Board will publish a Review on a Web page for application profiles. Each Review will include, at a minimum: any comments from the Usage Board on the application profile; pointers to locally archived copies of the application profile as originally submitted and (if necessary) as subsequently amended in light of Usage Board comments; a pointer (with appropriate disclaimers) to the "latest version" of an application profile held by its maintainers.

6.8. Persistent identifiers for reviewed Application Profiles. Review represents a form of recognition, and its URL will be persistent for purposes of citation.

7. New terms proposed with an Application Profile

7.1. Evaluation of new terms. New terms appearing in application profile submissions must be evaluated for compliance with the DCMI Abstract Model prior to evaluation of the Application Profile itself.

7.2. Assignment of DCMI term URIs and status. New terms deemed in compliance with the DCMI Abstract Model may be given URIs in DCMI namespaces and assigned a status of conforming.

7.3. Conformance criteria. Decisions as to whether a proposed term is in compliance with the DCMI Abstract Model will be made using the DCMI-Compliant Term Decision Tree.

8. Proposals for endorsement of terms in other namespaces for use within Application Profiles

8.1. Existing terms housed in other namespaces to be used within Application Profiles seeking review must be evaluated for compliance with the DCMI Abstract Model prior to evaluation of the Application Profile itself.

9.

Revisions of existing DCMI terms

9.1. Proposals for revisions. Requests to change terms in DCMI namespaces may originate within the Usage Board or externally. A Usage Board member will be assigned to draft a proposal for a change. Changes provisionally approved by the Usage Board will be circulated for general comment on the DC-General discussion list for one month before final approval. Final approval for term changes without significant opposition may be approved by email or teleconference vote.

9.2. Changes for formally standardized terms. Terms from namespace http://purl.org/elements/1.1/ require changes to the relevant standards: ISO Standard 15836-2003 (February 2003) and NISO Standard Z39.85-2001 (September 2001).

9.3. Terms from DCMI-hosted namespaces [to be added]

9.4. Application profile terms residing on DCMI hosted namespaces will be subject to the same change processes as other DCMI terms, but managed by the entities responsible for the terms. Application profile terms residing on non-DCMI namespaces will be subject to term policies of the host entity.

9.5. Changes to already 'conforming' application profiles require further Usage Board review of the application profile in whole or in part according to the processes and criteria outlined in previous sections. Changes to DCMI-registered "conforming" application profiles will be versioned according to DCMI namespace policies.

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

REVIEW OF PROCESS DOCUMENT (http://dublincore.org/usage/documents/process/)
Stuart Sutton

2007-03-12

SECTION 6.1. Application Profiles Subject to Review

We still do not have a definitive definition of a "DCMI Strategic Activity." It is framed as the primary mechanism for determining which initiatives will be bringing application profiles to the Usage Board.

SECTION 6.2. Documentation of Application Profiles

The section states that the applicant must include "an identifier of the element set where [the term] is defined, ideally in the form of URIs for individual terms." Since the terms have to conform to the Abstract Model, don't we require "URIs for the individual terms"?

SECTIONS 6.4 through 6.8 are numbered incorrectly. There is no section 6.3.

SECTION 6.5. Evaluation of terms in Application Profiles.

The section speaks of the "use of terms related to Dublin Core ... will be evaluated from the standpoint of semantic conformance, grammatical principle (eg, 'dumb-down'), clarity, and good practice." Doesn't this mean "terms related to Dublin Core" and not "use of terms related to Dublin Core"?

SECTION 6.8. Persistent identifiers for reviewed Application Profiles.

The section states: "Review represents a form of recognition, and its URL will be persistent for purposes of citation." Which URL? The Application Profile or the Usage Board review?

SECTION 7.2. Assignment of DCMI term URIs and status.

This section states: "New terms deemed in compliance with the DCMI Abstract Model may be given URIs in DCMI namespaces and assigned a status of conforming." Does this really mean "may be given," or is it "will be given." If it is "may be given," what's the criteria for being in?

SECTION 8. Proposals for endorsement of terms in other namespaces for use within Application Profiles.

This section says that terms from other name spaces in application profiles need to be reviewed for conformance to the Abstract Model. The section actually does not say anything about assigning a status to those terms; BUT, the section title speaks of "endorsement" which seems wrong to me. Endorsement (as defined in section 3) speaks of endorsing "assertions" by another that a term conforms (e.g., the LOC relators) and not that the term itself conforms. It seems to me that what we are talking about here in this section is "conforming" and not "endorsed."

Title: FOAF
Identifier: http://stage.dublincore.org/usage/meetings/2007/08/singapore/.html/foaf.html
Created: 2007-07-19

2007-08-18 From: Andrew Wilson <andrew.wilson@NAA.GOV.AU>

Couple of things about the meeting packet. I inadvertently reversed the order of references 7 and 8. Tom, could you please change them around on the page at: http://stage.dublincore.org/usage/meetings/2007/08/singapore/.html/foaf.html. Ie. The URL at reference 7 should actually be at reference 8 and vice versa. Thanks.

I know I gave quite a lot of references for this topic, probably too much for people to read in practice. I'm not sure that there is much point in actually including the FOAF specification as I suspect people won't have time to read it. Maybe Tom can just provide the URL without actually including the document in the meeting packet? Can we include in the packet the draft functional requirements document, the assessment of FOAF against the FR, and Dan's short paper on describing agents with FOAF, as well as the excerpt from the ePrints application profile referred to by Andy and which is included in Tom's first email about the meeting packet? I think that should be more than sufficient.=20

Re Topic 4:

This is about exploring the viability of using FOAF as the DCMI recommendation for describing agents associated with the resources described by DCMI description sets. I hope that at the UB meeting we can answer the following questions:
1. Is the Agents FR document adequate for the needs of the DCMI community?
2. How well does FOAF meet the Agents FR? Is the ePrints AP work relevant and useful?
3. Is FOAF/ePrints adequate and appropriate for describing agents as part of DC description sets?
4. If the answer to 3 is 'yes', what needs to be done to make FOAF/ePrints useable?

The basic minimum reading required for this topic is the Agents FR draft [1], Dan's assessment of FOAF against the draft FR [2], and the relevant section of the ePrints AP [3]. It would be really helpful if the UB could also read Dan's introduction to using FOAF [4] but I think we can manage without it if they don't.

[1] http://dublincore.org/groups/agents/agentFRdraft2-2.html
[2] http://dublincore.org/agentswiki/FoafReview
[3] http://www.ukoln.ac.uk/repositories/digirep/index/EPrints_Application_Profile#Description_of_an_Agent
http://stage.dublincore.org/usage/meetings/2007/08/singapore/Eprints-excerpt.pdf
[4] http://rdfweb.org/mt/foaflog/archives/2003/07/10/12.05.33/

2007-07-18 - from Andrew

The Agents Task Group, which consists of myself, Tom Baker, and Dan Brickley, was set up following DC2006 to progress work on developing a method for describing Agents as part of DC description sets. There are a number of unresolved work items waiting to be completed and Tom and I think the best chance to make progress would be to discuss the Friend Of A Friend (FOAF) and Agents requirements at the UB meeting in Singapore. It would be a great help if you could all come to Singapore prepared to discuss the Agent description issues. The Agents TG has its own Wiki [1] which includes the workplan and links to existing documents.

I'd like to set aside 60 minutes on the Singapore UB Agenda to discuss the agents work. The discussion must start from the basic position that it is not feasible for DCMI to develop a new set of properties to describe Agents. Although there is a lot of interest in the Community in how to describe agents there has been virtually no interest shown in contributing to the work of the Group. So, if the UB decide that FOAF

does not meet our agent description requirements then we will need to seriously consider the future of our agents work.

Before the older Agents Working Group was dissolved [2] we agreed that the best way forward was to develop an AP using existing properties [3]. Dan Brickley has been interested for some time in moving FOAF closer to the DCMI community and proposed that it be investigated by DCMI as a metadata schema for describing agents in DC description sets. Dan has recently done a comparison [6] of the Agents requirements, as articulated some years ago [4], and the FOAF specification [5]. A brief discussion about identifying things using FOAF at [7], written a few years ago by Dan B., will also be a good introduction to some of the issues. These documents should prepare you discuss whether FOAF is an adequate and appropriate mechanism for describing agents as part of DCMI description sets. FOAF is purely an RDF vocabulary so we need to discuss what consequences this would have for implementors should we agree to recommend FOAF.

It is important to understand how agent descriptions are linked to resource descriptions in description sets. A very helpful presentation on 'Agents and the DCAM' was given by Andy Powell to the Agents WG at the Madrid Conference [8].

Finally, it might be informative if you have time to also consider the work that IFLA is doing on Authority files in the draft Functional requirements for Authority data [9].

My hope is that we finish the UB meeting with agreement on the following issues:

1. Is the Agents FR document adequate?
2. How well does FOAF meet the Agents FR?
3. Is FOAF adequate and appropriate for describing agents as part of DC description sets?
4. If the answer to 3 is 'yes', what needs to be done to make FOAF useable?

[1] http://dublincore.org/agentswiki/FrontPage
[2] http://dublincore.org/groups/agents/
[3] http://www.jiscmail.ac.uk/cgi-bin/webadmin?A2=ind0510&L=dc-agents&P=154
[4] http://dublincore.org/groups/agents/agentFRdraft2-2.html
[5] http://xmlns.com/foaf/spec/
[6] http://dublincore.org/agentswiki/FoafReview
[7] http://rdfweb.org/mt/foaflog/archives/2003/07/10/12.05.33/
[8] http://dublincore.org/groups/agents/dc2005_dc-agents-meeting.ppt
[9] http://www.ifla.org/VII/d4/FRANAR-ConceptualModel-2ndReview.pdf

DRAFT DRAFT DRAFT DRAFT DRAFT DRAFT Functional Requirements for Describing Agents

Dublin Core Metadata Initiative - Agents Working Group

Date: 30 January 2004

Creator: Andrew Wilson, andrewcwilson@ozemail.com.au
National Archives of Australia
Contributor: Robina Clayphan robina.clayphan@bl.uk
British Library

Status of this document: *Working Draft*

Change history: *Draft 2* (2004-02-05)

Description: This document outlines a set of functional requirements for describing agents.

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

1. Background/Discussion

There is some ambiguity with this issue. The principle question is whether we are trying to 'describe' agents or 'identify' them? How relevant or important is the question? Dublin Core metadata is used for descriptions of resources for the purposes of making discovering them easier. Therefore we characterise DC metadata records as description for discovery. So can we apply this concept to agent descriptions? Perhaps we are describing agents for the purpose of unambiguously identifying them so they can be correctly associated with the resources for which they are responsible? In other words, description for identification. Agent descriptions, therefore, serve two purposes: description *and* identification. So we are trying to describe agents in a way that will allow us to:

- **disambiguate different agents who have shared or similar attributes (such as name, etc);**
- **recognise when agents are the same, despite appearing to be different, for example different presentations of the same name, pseudonyms, etc.;**
- **contact the correct agent associated with a resource;**
- **and collocate all the works of any specific agent.**

Disambiguation may be the most significant of these purposes. It enables effective searching for resources by enabling a reasonable degree of certainty about associated agents, and it is essential for protection of intellectual property and to assist with copyright payments, where a high degree of certainty about agents is needed.

So the resource description/discovery community needs an agent core because the DC element set does not allow a sufficiently precise description of an agent to support the above functions.

2. Scope

This document aims to set out the requirements and the metadata elements needed for unambiguously describing OR identifying the agents associated with resources. Agent descriptions may be contained within DC metadata records, or linked to the DC metadata records for particular resources as an associated metadata description. It is not within the scope of this document to consider the issue of where agent descriptions should be located. The functional requirements set out in this document will form the basis for development of a core set of metadata elements for describing agents.

For the purposes of this document agents are defined as *persons* (author, publisher, sculptor, editor, director, etc.) or *groups* (organization, corporation, library, orchestra, country, federation, etc.) that have a role in the lifecycle of a resource.

We also point out the constraints of the various data protection acts which ensure that there is only a limited amount of data that can legally be recorded about persons. So dates and location may be problematic for living people unless their explicit permission to include such data is obtained.

3. Entities

We define two classes of agents in this document:

1. Person: an individual human being, living or dead; and

2. Group: a set, either existing or defunct, of individual entities acting collectively.

4. Attributes

Each class of entity has associated with it a set of attributes or characteristics that serve to identify that entity unambiguously from all other entities of either class.

4.1 Attributes of a Person

This document defines the attributes of a *person* as the following:

identifier
name
dates
title
affiliation

location
email
other information

1. **Identifier**

A scheme, numeric or alphabetic, or a combination of the two, used to identify unambiguously a specific individual agent. No such schemes yet exist. This element will allow for the use of such schemes when and if they are developed.

4.1.2 Name

The name or names by which the person is known, including alternative names.

4.1.3 Dates

May include date of the person's birth and/or death, or *floruit* dates (ie. an indication of the period in which the person was known to be active in a given field of endeavour).

4.1.4 Title

A word or phrase used to identify the rank, office, nobility, honour, etc. of the person.

4.1.5 Affiliation

The name of the organization, institution, company, or other body with which the person was or is associated, or by whom the person was employed or contracted.

4.1.6 Location

Information about the person’s principal area of residence over time. Context may be indicated by the use of appropriate qualifiers (for example: Lived in Canberra 1991-2005).

4.1.7 Email

Email address or addresses currently assigned to the person at the time of the description.

4.1.8 Other Information

Any additional significant information about the person that is needed to unambiguously identify that person.

4.2 Attributes of a Group

This document defines the attributes of a *group* as the following:

legal number
name
jurisdiction
location
dates
web site
other information

4.2.1 Legal number

Any official number assigned by a public authority that is used to identify the group.

4.2.2. Name

Names by which the group is or was known. May include other forms of the name and changes of name over time.

4.2.3 Jurisdiction

The legal name of the judicial and administrative entity which has jurisdiction over the territory in which the group operates.

4.2.4 Location

The place from which the group operated.

4.2.5 Dates

Dates indicating the period the group operated. May include such things as date of founding and dissolution, date of legal mandate establishing the group, etc.

4.2.6 Web Site

The http address of the world wide web site operated by the group.

4.2.7 Other Information

Any additional significant information about the group that is needed to unambiguously identify that group.

2007-08-25 Usage Board meeting, Singapore

This review of FOAF against the Agent Description Requirements (http://dublincore.org/groups/agents/agentFRdraft2-2.html) has been prepared by Dan Brickley and Tom Baker. Comments are welcome. There is a comment space provided at the end of the review. Please use feel free to use it.

Notes on FOAF and DC

This document addresses an action on DanBrickley and TomBaker to "investigate and assess feasibility of using FOAF for DC agent descriptions".

FOAF and the draft DC Agents requirements

Background

The Dublin Core community have periodically addressed the topic of Agent description for over a decade. While the original 15 DC terms were historically regarded as being for the description of (broadly) "document-like" objects, the need in the DC community to describe agents associated with those objects has been clear since the early days of the initiative. For example, at the Dublin Core Workshop in Helsinki, in October 1997, UKOLN/AHDS work on deploying DC was presented. Their report, Discovering Online Resources. Unifying Resource Discovery Metadata for the Humanities: An Application Based Upon the Dublin Core articulated some common expectations from the metadata community at the time. For example, properties such as "personalName", "corporateName", "affiliation", "email", and constructs for postal addresses, phone numbers, fax etc were required. It was generally recognised that these kinds of property were both useful, needed ... but yet not really properties of the core "document-like" object being described in some Dublin Core record.

This tension between application needs and practical scoping of the DC was central to DC discussions and decisions over the last decade. Dublin Core took as a scoping constraint the ideal that community extensions and refinements should in some sense be "optional" and "dumb-downable", so that all applications and users of DC would at least share some commom base understanding of a DC-based record. If DC were to take a very liberal approach to extensibility, allowing community refinements/qualifiers for DC terms which really encoded properties of various related resources (buildings, telephones, people), ... the core value of DC could be endangered. Intuitively, the phone number of some contributor to some document is not always usefully considered to be a direct property of that document. To avoid "polluting" the values of the 15 core DC terms with such related-but-different metadata, the DC community articulated the "1 to 1" principle, which was an attempt to formalise this constraint to guide the development and community extension of Dublin Core.

At the same time that these discussions were taking place, W3C's RDF initiative was maturing. First publically presented at the October 1997 DC Workshop in Helsinki, RDF provided a clean and formally-grounded model for describing arbitrary properties of arbitrary things, using a data model that made clear their inter-relationships and allowed for many potential ambiguities to be avoided.

From 2000, the FOAF project began experimenting with the use of RDF for describing people and agents in the Web. Unlike DC, organizationally FOAF was an informal collaboration amongst members of the RDF and Semantic Web developer community. While standards-based, it had a more experimental style and adopted the strategy of trying various terms in public to see which were adopted, and documenting them often only in retrospect. FOAF was designed as a pure RDF vocabulary, and as such expected to be used alongside other RDF vocabularies such as DC. The vocabulary evolution model of FOAF was itself derrived from the DC experience, ... in particular the decision to version at a term level rather than at the namespace level came after observing the deployment issues around DC's move from a 1.0 to a 1.1 namespace. The initially ad-hoc approach to FOAF deployment has created some legacy around some parts of the FOAF design, in particular the properties for structured names.

As of May 2007, the FOAF specification defines 12 classes and 53 properties. Each of these is assigned a "term status" of "stable", "unstable" or "testing", corresponding to a set of (currently loosly defined) expectations about likelihood of further change. This term status vocabulary is also used at W3C by the SKOS project within the Semantic Web Deployment Working Group, and in future versions of FOAF is likely to be complimented by use of the OWL "Depracted" construct, for terms (eg. "geekcode") whose status in the spec is likely to be downplayed due to lack of adoption or frivolous nature. Discussions are in progress, facilitated by W3C's Semantic Web Coordination Group, regarding the problem of long-term persistence of the FOAF namespace, and in particular the xmlns.com domain name that it depends upon. Mechanisms being discussed including sharing of the DNS registrar password with a group of advisors, periodic "heartbeat" progress reporting to bodies such as W3C SWCG and DCMI, and offsite archival of relevant documentation.

The current version of the FOAF specification (in HTML and RDF) can be found at http://xmlns.com/foaf/spec/. At each revision, a dated snapshot is published and archived. Within each version of the specification, hypertext anchors are available for "permalinks" to each description of each FOAF term. This design allows for the vocabulary to evolve gradually, with a historical record of former documentation always being available. A full CVS version history of the files is also kept.

The remainder of this document assesses the utility of the FOAF specification with respect to the DCMI community's draft Agent requirements document.

1. Background/Discussion

```
There is some ambiguity with this issue. The principle question is
whether we are trying to ‘describe’ agents or ‘identify’ them?
```

Dublin Core is focussed on discovery, yet discovery is a hard concept to use for vocabulary scoping, since any property of any related object can in principle aid discovery. In particular, as noted in the Agents requirements, the identification of agents through the provision of agent descriptions can support discovery. FOAF similarly combines a concern for agent description with agent discovery. In particular, the FOAF discussions have technical design have had a strong emphasis on the need for formally grounded, flexible and pluralistic approaches to agent identification. To accomplish this, FOAF makes use of the W3C OWL language, and defines certain FOAF properties as "inverse functional" properties. For example, the foaf:homepage, foaf:mbox and foaf:isPrimaryTopicOf properties are considered "inverse functional". Technically, this means that there can be at most one thing in the world that has any given value for one of these properties. Terms such as foaf:homepage, foaf:weblog and various Instant Messager properties address the problem of identifying "modern" Web users; the foaf:mbox_sha1sum term is a quirky but widely used mechanism for indirectly identifying people in terms of a number derrived unambiguously from the address of a mailbox they are the primary owner of. In addition, the foaf:primaryTopic property and its inverse, foaf:isPrimaryTopicOf allow for the indirect identification of agents through describing them in terms such as "the person thatis the primary topic of the document whose URI is http://en.wikipedia.org/wiki/Isambard_Kingdom_Brunel".

Using such techniques, FOAF provides an approach to identification-by-description that is (a) formally grounded, in terms of the semantics provided by OWL and RDFS (b) extensible, in that the same techniques can be used with new terms as they emerge, whether in FOAF or from other namespaces (c) pluralistic: a typical FOAF description can use any combination of identifiers and reference-by-description techniques. The FOAF identification approach is designed to be consistent with Web Architecture and allows

for (but does not require) the use of URI identifiers for people and other agents. The document "identifying things in FOAF" describes the approach taken in a little more detail.

The DC Agent requirements goes on to note that "Agent descriptions, therefore, serve two purposes: description and identification.", identifying the following specific purposes of identifying agents.

- disambiguate different agents who have shared or similar attributes (such as name, etc);
- recognise when agents are the same, despite appearing to be different, for example different presentations of the same name, pseudonyms, etc.;
- contact the correct agent associated with a resource;
- and collocate all the works of any specific agent.

FOAF addresses these requirements, to the extent possible given some specific dataset. Basic identity reasoning can be conducted purely by following the semantics of the OWL constructs used (inverse functional etc). Richer (and less formally guaranteed) disambiguation strategies can also be used. Two FOAF descriptions, for example, might mention each a person called "John Smith" who was born on the same day, and who works for the same corporation. FOAF allows this commonality to be expressed, yet doesn't offer any formal guarantee that the descriptions are in fact describing the same person. This is perhaps likely, probably, yet not implied by the meaning of the terms used in the description. W3C's RDF querying language, SPARQL, can be used to express matches such as these - for example finding entries in a database of people descriptions where properties such as name, birthday and workplace match.

The Agent requirements draft argues:

```
So the resource description/discovery community needs an agent core because the
DC element set does not allow a sufficiently precise description of an agent to support the above functions.
```

A FOAF perspective here might be slightly different; a little more "meta". Rather than there just being a core of properties for describing people when identifying people, we also need some higher-level strategies, such as the use of OWL's "inverse functional" mechanism, to allow additional properties from other parties to be acknowledged as uniquely identifying. Having said that, a lot can be done with the basic properties defined in FOAF, in particular primaryTopic can link a person to the ID of a document that is known to identify them.

Scope

```
This document aims to set out the requirements and the metadata elements needed for unambiguously describing OR
identifying the agents associated with resources. Agent descriptions may be contained within DC metadata records,
or linked to the DC metadata records for particular resources as an associated metadata description. It is not
within the scope of this document to consider the issue of where agent descriptions should be located. The
functional requirements set out in this document will form the basis for development of a core set of metadata
elements for describing agents.
```

These constraints are consistent with the FOAF design. As an RDF vocabulary, FOAF descriptions can be mixed, partitioned and inter-linked quite freely.

```
For the purposes of this document agents are defined as persons (author, publisher, sculptor, editor, director,
etc.) or groups (organization, corporation, library, orchestra, country, federation, etc.) that have a role in the
lifecycle of a resource.
```

FOAF defines a term, foaf:Agent as well as a short, non-exhaustive list of sub-classes of Agent. These are:

* foaf:Person * foaf:Organization * foaf:Group

FOAF does not currently define detailed terms such as "sculptor". Instead, the expectation is that lists such as the MARC relator terms would be used.

FOAF does define one specific relationship in this area: foaf:maker (and an inverse, foaf:made). The foaf:maker property relates something to a foaf:Agent that foaf:made it. The FOAF specification currently recommends that dc:creator be used only for simple string values. This recommendation should be updated as the DCAM and RDF encoding are finalised. There is an entry in the FOAF wiki on the motivation for defining foaf:maker; briefly, it was created to ensure a simple, regular construct that did not have as many deployment variations as dc:creator, to lower the burden on applications that encounter the property.

```
We also point out the constraints of the various data protection acts which ensure that there is only a limited
amount of data that can legally be recorded about persons. So dates and location may be problematic for living
people unless their explicit permission to include such data is obtained.
```

As an RDF vocabulary (rather than e.g. an XML format), FOAF does not make mandatory the inclusion any particular information. It defines the meaning of terms, rather than the required content of documents. Consequently it can be used differently in different institutional or legal settings.

3. Entities

```
We define two classes of agents in this document:
```

```
1. Person: an individual human being, living or dead; and

2. Group: a set, either existing or defunct, of individual entities acting collectively.
```

These correspond well to foaf:Person and foaf:Group. In FOAF, a Group is a group of Agents rather than necessarily of Persons. Furthermore, a Group is itself an Agent, and can therefore be used (where appropriate) within FOAF descriptions wherever an Agent is expected. FOAF provides some technical machinery (again based on W3C OWL) for characterising the membership criteria for a Group based on their properties (defined using RDF terms). This aspect of FOAF is likely to evolve to make better use of new technology under development at W3C (eg. RIF rules, SPARQL queries, OWL 1.1).

4. Attributes

```
Each class of entity has associated with it a set of attributes or characteristics that serve to identify that
entity unambiguously from all other entities of either class.
```

FOAF, as an RDF vocabulary, has the notion of "property" at its heart. In RDF, properties are defined in terms of the classes they make sense to be used with, ... rather than a class defining in any exhaustive or centralised way the list of properties/attributes it expects. In practical terms, we can read "attribute" in the Agent requirements as "property" in the FOAF/RDF sense with no loss of meaning.

4.1 Attributes of a Person

```
This document defines the attributes of a person as the following:
identifier
name
dates
title
affiliation
location
email
other information
```

```
4.1.1. Identifier
A scheme, numeric or alphabetic, or a combination of the two, used to identify unambiguously a specific individual
agent. No such schemes yet exist. This element will allow for the use of such schemes when and if they are
developed.

4.1.2 Name
The name or names by which the person is known, including alternative names.

4.1.3 Dates
May include date of the person's birth and/or death, or floruit dates (ie. an indication of the period in which the
person was known to be active in a given field of endeavour).

4.1.4 Title
A word or phrase used to identify the rank, office, nobility, honour, etc. of the person.

4.1.5 Affiliation
The name of the organization, institution, company, or other body with which the person was or is associated, or by
whom the person was employed or contracted.

4.1.6 Location
Information about the person’s principal area of residence over time. Context may be indicated by the use of
appropriate qualifiers (for example: Lived in Canberra 1991-2005).

4.1.7 Email
Email address or addresses currently assigned to the person at the time of the description.

4.1.8 Other Information
Any additional significant information about the person that is needed to unambiguously identify that person.
```

4.2 Attributes of a Group

```
This document defines the attributes of a group as the following:

legal number
name
jurisdiction
location
dates
web site
other information
```

4.2.1 Legal number
Any official number assigned by a public authority that is used to identify the group.

4.2.2. Name
Names by which the group is or was known. May include other forms of the name and changes of name over time.

4.2.3 Jurisdiction
The legal name of the judicial and administrative entity which has jurisdiction over the territory in which the group operates.

4.2.4 Location
The place from which the group operated.

4.2.5 Dates
Dates indicating the period the group operated. May include such things as date of founding and dissolution, date of legal mandate establishing the group, etc.

4.2.6 Web Site
The http address of the world wide web site operated by the group.

4.2.7 Other Information
Any additional significant information about the group that is needed to unambiguously identify that group.

COMMENTS

[If you have any comments, views, opinions about the review please add them here]

http://purl.org/dc/terms/license

Definition

A legal document giving official permission to do something with the resource.

Eprint-specific recommendation

The terms under which the described copy of a manifestation of an expression of the eprint is made available. Typically, the URI of a licence statement should be provided as a *value URI*.

For example:

```
Statement (
  Property URI ( dcterms:license )
  Value URI ( http://creativecommons.org/licenses/by/2.0/ )
)
```

Date Available [edit]

---

http://purl.org/dc/terms/available

Definition

Date (often a range) that the resource will become or did become available.

Eprint-specific recommendation

The date that the described copy of a manifestation of an expression of the eprint will become or did become public.

Use a *value string* to provide the date, formatted according to the W3C Date Time Format (W3CDTF) specification.

For example:

```
Statement (
  Property URI ( dcterms:available )
  Value String ( "2004-09-23"
    Syntax Encoding Scheme URI ( dcterms:W3CDTF )
  )
)
```

Is Part Of [edit]

---

http://purl.org/dc/terms/isPartOf

Definition

The described resource is a physical or logical part of the referenced resource.

Eprint-specific recommendation

A collection, typically a bibliographic collection, of which the described copy is a member.

Use both a *value URI* and a *value string* to indicate the collection.

For example:

```
Statement (
  Property URI ( dcterms:isPartOf )
  Value URI ( <http://purl.org/poi/iesr.ac.uk/1084445801-13323> )
  Value String ( "JSTOR" )
)
```

Description of an Agent [edit]

2007-08-25 Usage Board meeting, Singapore

Entity type [edit]

---

http://purl.org/dc/elements/1.1/type

Definition

The type nature or genre of the content of the resource.

Eprint-specific recommendation

Each of the entity *descriptions* in the *description sets* conforming with this application profile will need to be explicitly typed. This is done using a dc:type *statement* with one of the following *value URIs* taken from the Eprints EntityType Vocabulary Encoding Scheme corresponding to the entity being described.

For example:

```
Statement (
  Property URI ( dc:type )
  Vocabulary Encoding Scheme URI ( eprint:EntityType )
  Value URI ( <http://purl.org/eprint/entityType/Person> )
)

Statement (
  Property URI ( dc:type )
  Vocabulary Encoding Scheme URI ( eprint:EntityType )
  Value URI ( <http://purl.org/eprint/entityType/Organization> )
)
```

Name [edit]

---

http://xmlns.com/foaf/0.1/name

Definition

A name for some thing.

Eprint-specific recommendation

A name for the agent (person or organisation).

In general, use foaf:family_name and foaf:givenname for describing persons and foaf:name for describing organisations.

Where a name is provided, see #A note about the form of personal and organisational names used in value strings.

For example:

```
Statement (
  Property URI ( foaf:name )
  Value String ( "University of York" )
)
```

Family Name [edit]

---

http://xmlns.com/foaf/0.1/family_name

Definition

The family name of some person.

Eprint-specific recommendation

The family name of a person.

In general, use foaf:family_name and foaf:givenname for describing persons and foaf:name for describing organisations.

For example:

```
Statement (
  Property URI ( foaf:family_name )
  Value String ( "Powell" )
)
```

Given Name [edit]

---

http://xmlns.com/foaf/0.1/givenname

Definition

The given name of some person.

Eprint-specific recommendation

The given name of a person.

In general, use foaf:family_name and foaf:givenname for describing persons and foaf:name for describing organisations.

For example:

```
Statement (
  Property URI ( foaf:givenname )
  Value String ( "Andy" )
)
```

Workplace Homepage [edit]

---

http://xmlns.com/foaf/0.1/workplaceHomepage

Definition

A workplace homepage of some person; the homepage of an organization they work for.

Eprint-specific recommendation

The homepage of the organisation for which the author of the eprint works.

Use this *property* to provide the URI of the organisational homepage as a *value URI*.

For example:

```
Statement (
  Property URI ( foaf:workplaceHomepage )
  Value URI ( <http://www.bristol.ac.uk/> )
)
```

Mailbox [edit]

---

http://xmlns.com/foaf/0.1/mbox

Definition

A personal mailbox, ie. an Internet mailbox associated with exactly one owner, the first owner of this mailbox. This is a 'static inverse functional property', in that there is (across time and change) at most one individual that ever has any particular value for foaf:mbox.

Eprint-specific recommendation

A mailbox associated with a person, formatted as a 'mailto' URI.

2007-08-25 Usage Board meeting, Singapore

For example:

```
Statement (
  Property URI ( foaf:mbox )
  Value URI ( "mailto:fred@example.com" )
)
```

Homepage [edit]

---

http://xmlns.com/foaf/0.1/homepage

Definition
A homepage for some thing.

Eprint-specific recommendation

Provide the URI of the agent's Web homepage as a *value URI*.

For example:

```
Statement (
  Property URI ( foaf:homepage )
  Value URI ( "http://www.bham.ac.uk/" )
)
```

Entity typing [edit]

Each of the entity *descriptions* in the *description sets* conforming with this application profile will need to be explicitly typed. This is done using a dc:type *statement* with one of the following *value URIs* taken from the Eprints EntityType Vocabulary Encoding Scheme:

- http://purl.org/eprint/entityType/ScholarlyWork
- http://purl.org/eprint/entityType/Expression
- http://purl.org/eprint/entityType/Manifestation
- http://purl.org/eprint/entityType/Copy
- http://purl.org/eprint/entityType/Person
- http://purl.org/eprint/entityType/Organization

For example:

```
Statement (
  Property URI ( dc:type )
  Vocabulary Encoding Scheme URI ( eprint:EntityType )
  Value URI ( <http://purl.org/eprint/entityType/Copy> )
)
```

A note about the form of personal and organisational names used in value strings [edit]

Where personal or organisational names are provided as *value strings*, the following guidelines should be followed.

Personal names should be listed surname or family name first, followed by comma-space, then the "usual" or preferred form of forenames and/or initials followed by a full stop. If necessary, this may be followed by the full set of spelled-out forenames in round brackets. Use the same form of name for all eprints, irrespective of the form used on the item itself.

For example:

```
Statement (
  Property URI ( marcrel:EDT )
  Value String ( "Bloggs, Fred" )
)

Statement (
  Property URI ( marcrel:EDT )
  Value String ( "Sulston, John E." )
)

Statement (
```

```
  Property URI ( marcrel:FND )
  Value String ( "Walker, J. J. (John Joseph)" )
)
```

In the case of organizations where there is clearly a hierarchy present, list the parts of the hierarchy from largest to smallest, separated by full stops. If it is not clear whether there is a hierarchy present, or unclear which is the larger or smaller portion of the body, give the name as it appears in the eprint.

For example:

```
Statement (
  Property URI ( marcrel:THS )
  Value String ( "Loughborough University. Department of Computer Science" )
)

Statement (
  Property URI ( dc:creator )
  Value String ( "University of Reading. Rural History Centre" )
)

Statement (
  Property URI ( dc:publisher )
  Value String ( "John Wiley & Sons, Inc. (US)" )
)
```

The inclusion of personal and corporate name headings from authority lists constructed according to AACR2, e.g. the Library of Congress Name Authority File (LCNA), is also acceptable.

A note about using identifiers

[edit]

Each of the entities in the model may be assigned a URI, encoded as the *resource URI* in the description of that entity.

It should be noted, however, that assigning and encoding a URI is not mandatory for any of the entities. Although the lack of a URI for any entity means that the entity can not be referenced from within metadata records in other eprint systems, the assignment of URIs to entities should be performed with care (and, where possible, after first checking whether a URI has already been assigned to the entity elsewhere). Multiple URIs for a single entity are likely to cause problems for downstream eprint systems, since it may be difficult for them to determine algorithmically whether any two URIs identify the same entity.

Where the ScholarlyWork and Expression entities in a *description set* have additional URIs, these should be encoded in one or more dc:identifier *statements*. As noted in the relevant sections above, in cases where there are multiple URIs for any entity, one of the URIs should be used as the *resource URI*. Recommended best practice is to use a DOI or Handle as the *resource URI*, where these exist.

The URIs assigned to each of the entities should uniquely identify a single entity. There may be more than one URI per entity, but each URI should only identify one entity.

It is an implementation issue whether the URIs assigned to the ScholarlyWork and Expression entities dereference to a representation of that entity. However, it is anticipated that eprint systems are likely to be configured such that the URI for ScholarlyWork and/or Expression entities dereference to a 'jump-off page' for the eprint, as served by the archive.

In all cases where a dc:identifier *statement* is provided, use a *syntax encoding scheme URI* to indicate that a URI is being provided.

For example:

```
Statement (
  Property URI ( dc:identifier )
  Value String ( "http://eprints.bath.ac.uk/archive/00000003/"
    Syntax Encoding Scheme URI ( dcterms:URI )
  )
)
```

It is not yet clear whether current practice is to assign DOIs at the level of the Work or at the level of the Expression (as those terms are used in FRBR) or in a more fine-grained way. The current Crossref documentation indicates that DOIs should be assigned to the 'work', however it is not clear that this usage of the word 'work' corresponds with its usage in FRBR - in fact, it is rather more likely that it corresponds with the use of the term Expression in FRBR.

There is also other evidence that publishers typically assign DOIs at the level of the FRBR Expression. For example, the DOI Handbook discussion about 'granularity' concludes that in practice it is safer to assign separate DOIs to each language translation of a given article, i.e. to assign DOIs at the level of the Expression.

As a result, it is suggested that the following guidelines are followed when making use of DOIs in the context of this application profile:

- If the DOI has been assigned to all possible expressions and manifestations of the eprint (e.g. it is intended to resolve to both the HTML format of the preprint and the PDF format of the formally published article), use the DOI as the *resource URI* for the ScholarlyWork entity (and also encode it as the *value string* of a dc:identifier *statement* for that entity).
- If the DOI has been assigned to all manifestations of a particular expression of the eprint (e.g. it is intended to resolve to all the formats of the Spanish language version of the formally published article), use the DOI as the *resource URI* for the Expression entity.

- If the DOI has been assigned to a particular manifestation of an expression of the eprint (e.g. it is intended to resolve only to the PDF format of the French-language translation of the formally published article), use the DOI as the *resource URI* for the Manifestation entity.
- If the DOI has been assigned to a particular copy of a manifestation of an expression of the eprint (e.g. it is only intended to resolve to the particular copy of an article as served by a particular Web server), use the DOI as the *resource URI* for the Copy entity.

In all cases, the 'http://dx.doi.org/10.100/12345' form of URI encoding should be used.

Note that publisher-assigned DOIs are highly likely to fall into cases 1 or 2.

A note about mapping the Eprints Application Profile to Simple DC [edit]

Mapping the Eprints Application Profile to Simple DC is available as a separate document.

The mapping enables software to 'dumb-down' (i.e. transform) a *description set* that conforms with this application profile to a *description set* that conforms with simple DC. In this context, 'simple DC' means a *description set* that comprises a single *description* that only uses the 15 *properties* in the Dublin Core Metadata Element Set.

It is worth noting that by using this mapping, a *description set* that complies with this application profile will be dumbed-down to form a *description* that complies with the Using simple DC to describe eprints document produced by the ePrints UK project.

It is also worth noting that the resulting simple DC *description* is about the eprint as a ScholarlyWork. While this is not the only approach to mapping this application profile to simple DC (for example, it would also be possible to map this application profile to a set of simple DC *descriptions* about each of the Copy entities) it fits well with the intended usage of this application profile in the OAI Protocol for Metadata Harvesting. In this case, each OAI item will have associated records that correspond to both the oai_dc format and an XML format based on this application profile.

The particular mapping discussed here and in the separate document does not preclude alternative mappings being generated in the future.

Examples [edit]

Example 1 [edit]

```
@prefix dc: <http://purl.org/dc/elements/1.1/> .
@prefix dcterms: <http://purl.org/dc/terms/> .
@prefix eprint: <http://purl.org/eprint/terms/> .

DescriptionSet (
  Description (
    Resource URI ( <http://eprints.gla.ac.uk/503/> )
    Statement (
      Property URI ( dc:type )
      Value URI ( <http://purl.org/eprint/entityType/ScholarlyWork> )
    )
    Statement (
      Property URI ( dc:title )
      Value String ( "Attempts to detect retrotransposition and de novo deletion of Alus and other dispersed
                      repeats at specific loci in the human genome" )
    )
    Statement (
      Property URI ( dcterms:abstract )
      Value String ( "Dispersed repeat elements contribute to genome instability by de novo insertion and unequal
                      recombination between repeats. To study the dynamics of these processes, we have developed
                      single DNA molecule approaches to detect de novo insertions at a single locus and Alu-mediated
                      deletions at two different loci in human genomic DNA. Validation experiments showed these
                      approaches could detect insertions and deletions at frequencies below 10(-6) per cell. However,
                      bulk analysis of germline (sperm) and somatic DNA showed no evidence for genuine mutant
molecules,
                      placing an upper limit of insertion and deletion rates of 2 x 10(-7) and 3 x 10(-7),
respectively,
                      in the individuals tested. Such re-arrangements at these loci therefore occur at a rate lower
                      than that detectable by the most sensitive methods currently available." )
    )
    # keywords to be added
    # Alu; deletion; dispersed repeats; insertion; recombination; retroposition
    Statement (
      Property URI (dc:creator )
      Value String ( "Hollies, C.R." )
    )
    Statement (
      Property URI (dc:creator )
      Value String ( "Monckton, D.G." )
    )
    Statement (
      Property URI (dc:creator )
      Value String ( "Jeffreys, A.J." )
    )
```

RDFWeb and Friend of a Friend (FOAF)

RDFWeb and the FOAF (Friend of a Friend) project .

« mobile meta? | Main | FOAF Contradictions »

July 10, 2003

Identifying things in FOAF

There is growing interest in FOAF and its relationship to various approaches to "identity management" on the Internet. The FOAF approach to all this is distinctly pluralistic, to the extent that you might not even notice that there is a FOAF way of dealing with identity. There aren't, for example, 'FOAF identifiers' as such, although there is certainly a FOAF approach to identifying things. So this is a first cut at writing up some of the as-yet-unarticulated design assumptions behind FOAF. A more user-friendly version would have examples, those will have to come later.

So here's the basic story. FOAF is built on top of W3C's Resource Description Framework (RDF), which itself uses XML and Unicode as file format standards. All FOAF documents are RDF documents, and any RDF application vocabularies (such as Dublin Core, RSS 1.0 core + extensions, MusicBrainz, Wordnet etc.) can be used within FOAF documents. FOAF shares with RDF a concern to use standard Web identifiers (URIs) wherever possible. The URI specification (RFC 2396) provides a common syntax for naming things on the Web, providing an umbrella concept which covers both 'URLs' and 'URNs'.

To the extent that everything we want to talk about has a well known URI, this solves all our problems. Lots and lots of things that we want to talk about do have URIs. There are URIs for Web pages, for mailboxes, for Java classes, for telephones, for ISBN-registered publications, and so on. This is great - when you want to talk about one of these things in a FOAF file, you just mention its URI. Simple, decentralised, standard.

However our story doesn't end here, FOAF needs to play in a world where we don't all have total knowledge of every relevant fact. Sometimes a thing might 'have' a URI (in some pedantic sense) yet 99% of parties on the Web might not know what that URI *is*. Or, closer to my main theme, we might want to talk in our FOAF files about things that it has proved peculiarly difficult to get agreement about identifying. People, for example.

Just try setting up a planet-wide system for identifying people and you'll see my point. There is significant resistence to the idea of creating a single set of identifiers used to 'tag' everyone. To put it mildly. So... where does this leave FOAF? FOAF documents are scattered around the Web, and each document makes a unique contribution to a bigger picture which can only be seen when those documents are merged together. In FOAF, we need to identify people, without there being agreement on person-identifiers. Tricky!

So here is the good news. RDF was designed for generic, cross-domain data merging. Imagine taking two arbitrary SQL databases and merging them, so that your new database could answer questions which required knowledge of things which were previously described partially in one dataset, and partially in another. That sort of operation is hard to do, because SQL wasn't designed in a way that makes this easy. Neither was XML. But RDF was, and FOAF is built as an RDF application. In RDF, there are off the shelf software tools which can take RDF documents, 'parse' them into a set of simple 3-part statements (triples) which make claims about the world, and store those statements alongside others in a merged RDF database. To the extent that both datasets use the exact same identifiers when mentioning things they describe, you get a rather handy data-merge effect.

So here is the (not very) bad news. If two different RDF files (eg. FOAF documents) are talking about the same thing but don't use exactly the same URI when mentioning that thing, how are our poor stupid computers supposed to be able to understand? In the real world, we want to write RDF documents (eg. for FOAF) about things that we've not yet agreed on common identifiers for. This is one of the core problems we've had to address in FOAF.

2007-08-25 Usage Board meeting, Singapore

Basically, off the shelf RDF tools can still do a lot to help us, but we have to help them. FOAF, as an application that focusses on the distributed, decentralised, *almost* out of control use of RDF 'in the wild', ran into this problem after we had about half a dozen FOAF files. There are now hundreds, soon thousands, of FOAF documents. Most of them talk about people, quite successfully, despite the absence of a global person-id registry. This sounds like a recipe for chaos, yet somehow many of our FOAF aggregation tools are quite happy with this situation. They can often figure out when two files are about the self-same thing, without much help from the authors of those documents. We do this using what might be called "reference by description". Instead of saying, "this page was created by urn:global-person-registry:person-n22314151", we say "this page was created by the peson whose (some-property...) is (some-value...)", taking care to use an unambiguous property such as foaf:homepage or foaf:mbox_sha1sum.

Here's how it works. Recall that FOAF is built on top of RDF, and so every FOAF document boils down to nothing more than a set of 3-part statements which relate two things together via terms such as 'workplaceHomepage', 'homepage', 'mbox'.

I am related to those things that are my homepages; FOAF's name for that relationship is 'foaf:homepage'.

I am related to those things that are my personal mailboxes by a relationship FOAF calls 'foaf:mbox'.

I am related to the strings that you get from feeding my mailbox identifiers to the SHA1 mathematical function by a relationship FOAF calls 'foaf:mbox_sha1sum'.

I am related to a myers briggs personality classification, FOAF calls that relationship 'foaf:myersBriggs'.

I am related to my workplace homepage (http://www.w3.org/) by a relationship called -- you guessed it -- 'foaf:workplaceHomepage'.

I am related to my name, 'Dan Brickley' by the 'foaf:name' relationship.

I am related to my AIM chat identifier by a relationship FOAF calls 'foaf:aimChatID'.

And so on. Other RDF vocabularies can define additional relationships (see the FoafVocab entry in our wiki for pointers). They all relate things to other things in named ways. A FOAF document, like any RDF document, is simply a collection of these simple claims about how things in the world relate.

But look again.There is a hidden pattern here. Some of these relationships are special.

foaf:homepage foaf:mbox foaf:mbox_sha1sum foaf:aimChatID fall in one category.

foaf:workplaceHomepage, foaf:myersBriggs, foaf:name fall in another.

Here's the difference. The former kinds of relationship (or 'property' in RDF-talk) have a special characteristic. They have been defined such that there is *at most one thing in the world* that has any particular value for that property.

There is... at most one thing in the world with any given foaf:homepage. Or foaf:mbox, or foaf:mbox_sha1sum, or foaf:aimChatID. By contrast, there may well be multiple things in the world with the same foaf:workplaceHomepage, or foaf:myersBriggs, or even (it's a big world) foaf:name. Apparently there's another Dan Brickley out there. And lots of my colleagues share my workplace homepage. And there are a lot of people who myers brigg surveys classify as 'INTP' . But there is nobody else at all who has the same foaf:homepage as me, or the same foaf:mbox. Or foaf:aimChatID.

This is one of the design principles underlying FOAF (and for that matter the entire Semantic Web effort): a pragmatic, pluralistic approach to resource description and identification. Rather than building big, centralised registries of people (or companies, or physical things) we look for cheaper, more lightweight shared strategies for identification. In FOAF, we do this by making sure there are multiple ways we can identify things.

So one FOAF file might mention 'here is a photo; it depicts the person whose mailbox is danbri@rdfweb.org'. Another

2007-08-25 Usage Board meeting, Singapore

FOAF file might say 'here is a weblog entry written by the person whose homepage is http://rdfweb.org/people/danbri/', a 3rd FOAF file might say, 'here is a chat transcript by the person whose foaf:aimChatID is danbri_2002'. To the extent that there is publically readable RDF in the Web that makes all these claims, *and* that there is, perhaps scattered around, enough information to deduce that these all describe the same people, RDF /FOAF tools can 'smush' it all together. They could 'realise' that the photo and the weblog and the chat log were all associated with the self-same thing, ie me.

To do that, we need certain pieces of information. We need to know which, of all the kinds of relationship there are, are the uniquely identifying ones. In RDF terminology we call these *unambiguous* (or more technically, *inverse-functional*) properties. When RDF software reads the FOAF spec it can determine this from markup embedded in the document itself. So machines can find out quite easily which properties are ones which uniquely identify people. They can do this for the FOAF spec, and for any other RDF vocabulary that is used alongside FOAF.

The other bit of information needed is that somewhere in the Web, it would need to be claimed that there is a person who has a mailbox of ... and a homepage of ... and an aimChatID of ...

If that information is available, then FOAF tools are all set to do the data merge, even though there is no planet-wide unified identification system for people. We don't use anything else except off the shelf standards: URIs plus W3C RDF and OWL technology.

If you find the data merging potential creepy, you are not alone. This kind of technology is not going away, but there are steps you can take. A full discussion of the privacy aspect isn't possible here, but the basic idea is (i) be aware -- scattered information can easily be merged (ii) keep things as secret as they need to be. Don't tell the world (in your FOAF file or elsewhere) all the chat IDs and homepages and mailboxes that you use, then act suprised when people and machines piece together your scattered contributions to the Web. Reading up on PGP might be a good idea.

We don't need to wait for a global identity management system before privacy and data merging becomes an issue. FOAF is intended to explore these issues, and to provide some advance warning for the way certain aspects of semantic web technology may affect our lives. Just as the world has had to adapt to the notion of 'being Googled' and having things that once seemed obscure now all to easily found, the rise of semantic web technology needs to be accompanied by an understanding of the risks and opportunities that 'being identified' presents.

Finally... a couple of points of further reading on the technical rather than social side of this problem. A couple of years ago I wrote a brief note on aggregation strategies which describes the 'smushing' problem. A more recent writeup by Matt Biddulph describing his Java implementation is worth a read too, as are many of the documents from the TAP project, which share FOAF's concern for reference-by-description. Guha and Rob's overview paper sets out the issues very clearly.

Posted by danbri at July 10, 2003 12:05 PM

Comments

Short version:

In FOAF, we use URIs to identify things while describing them.

When we don't have URIs handy, we take care to use identifying properties in our descriptions.

We don't care which properties we use, so long as they are unambiguous. The more that get into general use, the easier it becomes to figure out when two documents are describing the same entity.

Posted by: Dan Brickley on July 10, 2003 01:36 PM

2007-08-25 Usage Board meeting, Singapore

Coolio dude - thanks for this. But the process of smushing is kind of 'challenging' when the end-user holds onto their own foaf.rdf file.

So we have this idea of: a) hosting a database of foaf.rdf files that get 'shared' for FOAFster-type funtionality (if end-users don't care) and b) if they DO care about holding onto their own foaf.rdf files - then we assume that their file is the master, and we'll set up a mirroring process - where any changes to the master - is uploaded to the 'shared' database.

How does that sound?

Posted by: Marc Canter on July 10, 2003 04:12 PM

---

danbri, nice work.

Marc, I'm not sure the users holding their own files is likely to be such an issue. But to do stuff with the data you'll have to read it into your own system anyway. You could just store the URIs and poll them whenever needed, but it would probably make sense in terms of performance to keep a version in your DB as a cache. This should also lend itself to pretty simple load distribution - it doesn't really matter where the FOAF statements are held. So I think what you're suggesting probably would be a good approach, but for a slightly different reason ;-)

I know there's plenty of work being done with the query languages, but maybe systems like this call for a very simple standardised subset plus some system-level comms. i.e. just two or three fairly FOAF-specific queries (e.g. give me all your Persons with any of these properties) combined with URIs for the stores. Make it easy a possible to implement, but allow sync and the forwarding of queries between DBs (that may even have completely different purposes, but still expose the same mini-interface).

Posted by: Danny on July 11, 2003 01:17 AM

Great, and really illustrative example, Dan.

Marc and Danny: I am thinking about similar things, which I see as:

1. another level of DNS-like mapping to specific documents that gives those documents non-volitile identifiers

2. document stores that automatically plug-in to that DNS-like mapping, and provide interfaces exposing the stored documents to queries

3. tools for querying, caching, and syncing document data across different document stores

(4. safe and easy to integrate with existing/other tools, like websites, blogs, wikis, and email, and with other standards like N/echo and RSS!)

Posted by: Jay Fienberg on July 13, 2003 12:02 AM

Post a comment

Title: RDF declarations of DCMI terms
Identifier: http://stage.dublincore.org/usage/meetings/2007/08/singapore/.html/rdf-declaration-changes.html
Created: 2007-07-19

2007-06-17 Tom Baker

The finalization of a new version of the DCMI Abstract Model [1] and the development of a (not yet finalized) proposal to add domain and range declarations to DCMI properties [2] have consequences for how DCMI terms are declared in RDF and documented in Web pages.

In the RDF declarations:

1. Declarations such as
   <dc:type rdf:resource="http://dublincore.org/usage/documents/principles/#vocabulary-term"/>
   <dc:type rdf:resource="http://dublincore.org/usage/documents/principles/#encoding-scheme"/>
   <dc:type rdf:resource="http://dublincore.org/usage/documents/principles/#encoding-scheme"/>
   <dc:type rdf:resource="http://dublincore.org/usage/documents/principles/#element-refinement"/>
   change to
   <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Class"/>
   <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Datatype"/>
   <rdf:type rdf:resource="http://purl.org/dc/dcam/VocabularyEncodingScheme"/>
   <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property/>;

2. If it is convenient, terms formerly declared in the XML style
   <dcterms:DCMIType rdf:about="http://purl.org/dc/dcmitype/Event">
   <dcterms:DateScheme rdf:about="http://purl.org/dc/terms/W3CDTF">
   <dcterms:SubjectScheme rdf:about="http://purl.org/dc/terms/LCSH">
   <rdf:Property rdf:about="http://purl.org/dc/terms/bibliographicCitation">
   can be declared in the following style
   <rdf:Description rdf:about="http://purl.org/dc/dcmitype/Event">
   <rdf:Description rdf:about="http://purl.org/dc/terms/W3CDTF">
   <rdf:Description rdf:about="http://purl.org/dc/terms/LCSH">
   <rdf:Description rdf:about="http://purl.org/dc/terms/bibliographicCitation">
   because rdf:type is always specified (see point 1 above).

3. Members of Vocabulary Encoding Schemes, such as members of the DCMI Type Vocabulary, will be declared with an additional statement:
   <dcam:memberOf rdf:resource="http://purl.org/dc/terms/DCMIType"/>

4. Properties will have additional declarations of domains and ranges, e.g.:
   <rdfs:domain rdf:resource="http://example.org/dc/terms/BibliographicResource"/>
   <rdfs:range rdf:resource="http://example.org/dc/terms/BibliographicReference"/>

5. The Vocabulary Management Tool has hitherto supported the construction of classes such as DateScheme and FormatScheme, with URIs such as http://purl.org/dc/terms/DateScheme, in order to associate qualifiers (e.g., dcterms:W3CDTF) with terms qualified (in this case, dc:date). See for example [6]. These constructs are no longer supported by the abstract model, so the routines which create them should no longer be used. On the whole, this should lead to a simplification of the VMT scripts.

In the XML data used as input to the VMT (e.g., [7]), there will be several changes:

1. <Qualifies> statements will be deleted from new versions of terms. (These were used to create DateScheme, FormatScheme, etc.)

2. <Type-of-Term> will be updated, as above.

3. <Domain> and <Range> will be added.

The Web pages generated by the VMT will need to be updated:

1. Show domains and ranges for properties.

2. When Domain and Range classes are declared [2], the Web document will need a new section for these.

Beyond the RDF issues, there are implications
for the DCMI Registry as well, as the registry has been
customized on the basis of the pre-DCAM2007 typology of
terms and their URIs (e.g., "encoding scheme" and [8]).

[1] http://dublincore.org/documents/2007/06/04/abstract-model/
[2] http://dublincore.org/documents/2007/02/05/domain-range/
[3] http://dublincore.org/2006/12/18/dcq.rdf
[5] http://purl.org/dc/terms/DateScheme
[7] http://dublincore.org/usage/xml/terms.xml
[8] http://dublincore.org/usage/documents/principles/#encoding-scheme

Title: Domains and ranges
Identifier: http://stage.dublincore.org/usage/meetings/2007/08/singapore/.html/domains-ranges.html
Source: e:/work/dcub/singapore/index.txt
Created: 2007-07-19

Domains and ranges (Akira)

The Domain Vocabulary has been revised in light of Barcelona decisions [1] and posted for Public Comment through 30 July [2]. A range of rdfs:Literal has been proposed for dcterms:identifier [3]. Pete has written a note describing the "special cases" (title, date, description, and their subproperties) [4]. (From the notes: "There are two audiences for this comment: (1) multiple script communities and (2) SW community asking: best practice for unspecified ranges.")

[1] http://dublincore.org/usageboardwiki/DomainsActions
[2] http://dublincore.org/documents/2007/07/02/domain-range/
[3] http://dublincore.org/documents/2007/07/02/domain-range/#dctermsidentifier
[4] http://dublincore.org/usageboardwiki/RangesIssues

Reading

-- Domains and Ranges for DCMI Properties:
http://dublincore.org/documents/2007/02/05/domain-range/

-- Range Issues (Pete)
http://dublincore.org/usageboardwiki/RangesIssues

Comments

Comments are summarized at:
http://stage.dublincore.org/usage/meetings/2007/08/singapore/domains-digest1.txt
http://stage.dublincore.org/usage/meetings/2007/08/singapore/domains-digest2.txt

-- Bruce D'Arcus's comments with Pete's reply:
http://www.jiscmail.ac.uk/cgi-bin/webadmin?A2=ind0707&L=dc-architecture&T=0&F=&S=&P=413
http://www.jiscmail.ac.uk/cgi-bin/webadmin?A2=ind0707&L=dc-architecture&T=0&F=&S=&P=541
http://www.jiscmail.ac.uk/cgi-bin/webadmin?A2=ind0707&L=dc-architecture&T=0&F=&S=&P=680

1) It is problematic to use a property with both literal and non-literal values, from OWL-DL position and also from basic query and processing standpoint.
2) We should consider to separate properties for these cases.

-- Akira Miyazawa's comment
http://www.jiscmail.ac.uk/cgi-bin/webadmin?A2=ind0707&L=dc-architecture&T=0&F=&S=&P=3061

1) What most people think as a literal is not just a sequence of characters, but a text in some language.

2007-08-25 Usage Board meeting, Singapore

Dublin Core Metadata Initiative
Making it easier to find information

Domains and Ranges for DCMI Properties

Creator: Andy Powell
Eduserv Foundation, UK

Creator: Pete Johnston
Eduserv Foundation, UK

Creator: Thomas Baker
DCMI

Date Issued: 2007-07-02

Identifier: http://dublincore.org/documents/2007/07/02/domain-range/

Replaces: http://dublincore.org/documents/2007/02/05/domain-range/

Is Replaced By: Not applicable

Latest Version: http://dublincore.org/documents/domain-range/

Description of Document: This document describes a set of classes and specifies how those classes are used as domains and ranges of DCMI properties.

Table of contents

1. Introduction
2. The Classes
3. Domain and Range assertions for DCMI Properties
References

---

1. Introduction

This document uses the terminology of the DCMI Abstract Model [DCAM]. The relationship types with which this document is principally concerned are described by the DCAM as follows:

- Each *property* may be related to one or more *classes* by a *has domain* relationship. Where it is stated that a *property* has such a relationship with a *class* and a *described resource* is related to a *value* by that *property*, it follows that the *described resource* is an instance of that *class*.
- Each *property* may be related to one or more *classes* by a *has range* relationship. Where it is stated that a *property* has such a relationship with a *class* and a *described resource* is related to a *value* by that *property*, it follows that the *value* is an instance of that *class*.

In practice, this means that the *domain* indicates the *class* of *resources* that the *property* should be used to describe, while the *range* indicates the *class* of *resources* that should be used as *values* for that *property*.

The DCAM relationship types *has domain* and *has range* are the same as the RDF Schema [RDFS]

properties, `rdfs:range` and `rdfs:domain`.

2. Classes

This section describes a set of *classes* that are needed in order to describe *has domain* and *has range* relationships for DCMI *properties*.

The relationships between the classes are summarized in the UML class diagram in Figure 1:

Figure 1 - the Classes

The relationships between the classes associated with the Format property and its subproperties are particularly complex and are shown in more detail in Figure 2:

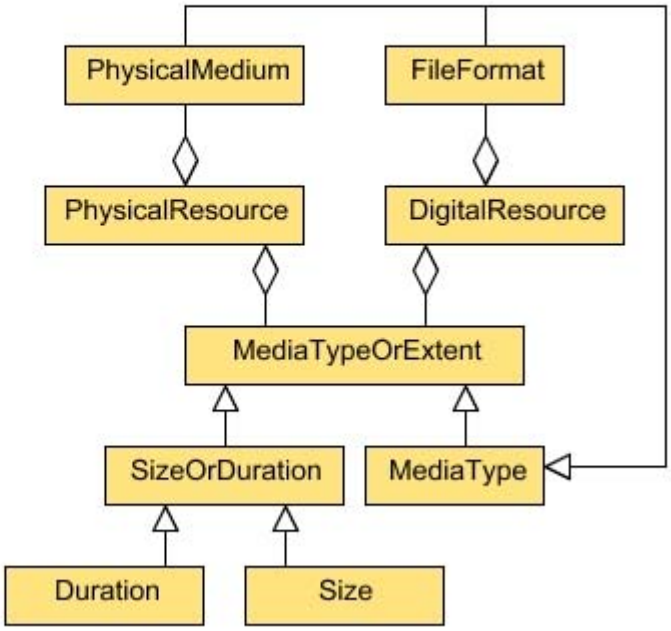


Figure 2 - the Classes associated with the Format property

2.1. Existing Classes Referenced in this Document

The terms described in this section are **existing** *classes* that have already been assigned URIs owned by DCMI or by other agencies, and for which term declarations are already made available by their owners. **No changes** to those term declarations are required or proposed.

Term Name: Resource	
URI:	http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]
Type of Term:	http://www.w3.org/2000/01/rdf-schema#Class [rdfs:Class]
Label:	Resource

Term Name: Literal	
URI:	http://www.w3.org/2000/01/rdf-schema#Literal [rdfs:Literal]
Type of Term:	http://www.w3.org/2000/01/rdf-schema#Class [rdfs:Class]
Label:	Literal

Term Name: Class

URI:	http://www.w3.org/2000/01/rdf-schema#Class [rdfs:Class]
Type of Term:	http://www.w3.org/2000/01/rdf-schema#Class [rdfs:Class]
Label:	Class

Term Name: Collection	
URI:	http://purl.org/dc/dcmitype/Collection [dcmitype:Collection]
Type of Term:	http://www.w3.org/2000/01/rdf-schema#Class [rdfs:Class]
Label:	Collection

2.2. Proposed Classes

The terms described in this section are **new** *classes* that are to be assigned URIs owned by DCMI. The URIs will be in the *DCMI Namespace* http://purl.org/dc/terms. A **new** set of term declarations will be made available, and those **new** term declarations are summarized in this section.

Term Name: Agent	
URI:	http://purl.org/dc/terms/Agent [dcterms:Agent]
Type of Term:	http://www.w3.org/2000/01/rdf-schema#Class [rdfs:Class]
Label:	Agent
Definition:	A resource that acts or has the power to act.
Comment:	Examples of an agent include a person, organization or software agent.
Instance Of:	http://purl.org/dc/terms/AgentClass [dcterms:AgentClass]

Term Name: AgentClass	
URI:	http://purl.org/dc/terms/AgentClass [dcterms:AgentClass]
Type of Term:	http://www.w3.org/2000/01/rdf-schema#Class [rdfs:Class]
Label:	Agent Class
Definition:	A group of agents.

Comment:	2007-08-25 Usage Board meeting, Singapore Examples of an agent class include students, women, charities, lecturers.
Sub-Class Of:	http://www.w3.org/2000/01/rdf-schema#Class [rdfs:Class]

Term Name: BibliographicResource	
URI:	http://purl.org/dc/terms/BibliographicResource [dcterms:BibliographicResource]
Type of Term:	http://www.w3.org/2000/01/rdf-schema#Class [rdfs:Class]
Label:	Bibliographic Resource
Definition:	A book, article or other documentary resource.

Term Name: DigitalResource	
URI:	http://purl.org/dc/terms/DigitalResource [dcterms:DigitalResource]
Type of Term:	http://www.w3.org/2000/01/rdf-schema#Class [rdfs:Class]
Label:	Digital Resource
Definition:	Something that is stored or transmitted as a sequence of discrete symbols from a finite set, usually binary data, represented using electronic or electromagnetic signals.

Term Name: Duration	
URI:	http://purl.org/dc/terms/Duration [dcterms:Duration]
Type of Term:	http://www.w3.org/2000/01/rdf-schema#Class [rdfs:Class]
Label:	Duration
Definition:	The time taken to view, play, execute, or interact with a digital resource or physical resource.
Comment:	For example, a duration might be a period in hours, minutes and seconds.
Sub-Class Of:	http://purl.org/dc/terms/SizeOrDuration [dcterms:SizeOrDuration]

Term Name: FileFormat	
URI:	http://purl.org/dc/terms/FileFormat [dcterms:FileFormat]

Type of Term:	http://www.w3.org/2000/01/rdf-schema#Class [rdfs:Class]
Label:	File Format
Definition:	A digital resource format.
Comment:	Examples include the formats defined by the the list of Internet Media Types.
Sub-Class Of:	http://purl.org/dc/terms/MediaType [dcterms:MediaType]

2007-08-25 Usage Board meeting, Singapore

Term Name: Frequency	
URI:	http://purl.org/dc/terms/Frequency [dcterms:Frequency]
Type of Term:	http://www.w3.org/2000/01/rdf-schema#Class [rdfs:Class]
Label:	Frequency
Definition:	A rate at which something recurs.

Term Name: Jurisdiction	
URI:	http://purl.org/dc/terms/Jurisdiction [dcterms:Jurisdiction]
Type of Term:	http://www.w3.org/2000/01/rdf-schema#Class [rdfs:Class]
Label:	Jurisdiction
Definition:	The extent or range of judicial, law enforcement, or other authority.
Sub-Class Of:	http://purl.org/dc/terms/LocationPeriodOrJurisdiction [dcterms:LocationPeriodOrJurisdiction]

Term Name: LicenseDocument	
URI:	http://purl.org/dc/terms/LicenseDocument [dcterms:LicenseDocument]
Type of Term:	http://www.w3.org/2000/01/rdf-schema#Class [rdfs:Class]
Label:	License Document
Definition:	A legal document giving official permission to do something with a Resource.
Sub-Class Of:	http://purl.org/dc/terms/RightsStatement [dcterms:RightsStatement]

2007-08-25 Usage Board meeting, Singapore

Term Name: LinguisticSystem	
URI:	http://purl.org/dc/terms/LinguisticSystem [dcterms:LinguisticSystem]
Type of Term:	http://www.w3.org/2000/01/rdf-schema#Class [rdfs:Class]
Label:	Linguistic System
Definition:	A system of signs, symbols, sounds, gestures, or rules used in communication.
Comment:	Examples include written, spoken, sign, and computer languages.

Term Name: Location	
URI:	http://purl.org/dc/terms/Location [dcterms:Location]
Type of Term:	http://www.w3.org/2000/01/rdf-schema#Class [rdfs:Class]
Label:	Location
Definition:	A named place or spatial region.
Sub-Class Of:	http://purl.org/dc/terms/LocationPeriodOrJurisdiction [dcterms:LocationPeriodOrJurisdiction]

Term Name: LocationPeriodOrJurisdiction	
URI:	http://purl.org/dc/terms/LocationPeriodOrJurisdiction [dcterms:LocationPeriodOrJurisdiction]
Type of Term:	http://www.w3.org/2000/01/rdf-schema#Class [rdfs:Class]
Label:	Location, Period or Jurisdiction
Definition:	A location, period or jurisdiction.

Term Name: MediaType	
URI:	http://purl.org/dc/terms/MediaType [dcterms:MediaType]
Type of Term:	http://www.w3.org/2000/01/rdf-schema#Class [rdfs:Class]
Label:	Media Type
Definition:	A file format or physical medium.

2007-08-25 Usage Board meeting, Singapore Sub-Class Of:	http://purl.org/dc/terms/MediaTypeOrExtent [dcterms:MediaTypeOrExtent]

Term Name: MediaTypeOrExtent	
URI:	http://purl.org/dc/terms/MediaTypeOrExtent [dcterms:MediaTypeOrExtent]
Type of Term:	http://www.w3.org/2000/01/rdf-schema#Class [rdfs:Class]
Label:	Media Type or Extent
Definition:	A media type or extent.

Term Name: MethodOfInstruction	
URI:	http://purl.org/dc/terms/MethodOfInstruction [dcterms:MethodOfInstruction]
Type of Term:	http://www.w3.org/2000/01/rdf-schema#Class [rdfs:Class]
Label:	Method of Instruction
Definition:	A process that is used to engender knowledge, attitudes and skills.

Term Name: MethodOfAccrual	
URI:	http://purl.org/dc/terms/MethodOfAccrual [dcterms:MethodOfAccrual]
Type of Term:	http://www.w3.org/2000/01/rdf-schema#Class [rdfs:Class]
Label:	Method of Accrual
Definition:	A method by which resources are added to a collection.

Term Name: PeriodOfTime	
URI:	http://purl.org/dc/terms/PeriodOfTime [dcterms:PeriodOfTime]
Type of Term:	http://www.w3.org/2000/01/rdf-schema#Class [rdfs:Class]
Label:	Period of Time
Definition:	An interval of time that is named or defined by its start and end dates.
Sub-Class Of:	http://purl.org/dc/terms/LocationPeriodOrJurisdiction [dcterms:LocationPeriodOrJurisdiction]

2007-08-25 Usage Board meeting, Singapore

Term Name: PhysicalMedium	
URI:	http://purl.org/dc/terms/PhysicalMedium [dcterms:PhysicalMedium]
Type of Term:	http://www.w3.org/2000/01/rdf-schema#Class [rdfs:Class]
Label:	Physical Medium
Definition:	A material or physical carrier of a physical resource.
Comment:	Examples include paper, canvas, etc.
Sub-Class Of:	http://purl.org/dc/terms/MediaType [dcterms:MediaType]

Term Name: PhysicalResource	
URI:	http://purl.org/dc/terms/PhysicalResource [dcterms:PhysicalResource]
Type of Term:	http://www.w3.org/2000/01/rdf-schema#Class [rdfs:Class]
Label:	Physical Resource
Definition:	A material thing.

Term Name: Policy	
URI:	http://purl.org/dc/terms/Policy [dcterms:Policy]
Type of Term:	http://www.w3.org/2000/01/rdf-schema#Class [rdfs:Class]
Label:	Policy
Definition:	A plan or course of action, as of a government, political party, or business, intended to influence and determine decisions, actions, and other matters.

Term Name: ProvenanceStatement	
URI:	http://purl.org/dc/terms/ProvenanceStatement [dcterms:ProvenanceStatement]
Type of Term:	http://www.w3.org/2000/01/rdf-schema#Class [rdfs:Class]
Label:	Provenance Statement
Definition:	A statement of any changes in ownership and custody of a resource since its creation that are significant for its authenticity, integrity and interpretation.

2007-08-25 Usage Board meeting, Singapore

Term Name: RightsStatement	
URI:	http://purl.org/dc/terms/RightsStatement [dcterms:RightsStatement]
Type of Term:	http://www.w3.org/2000/01/rdf-schema#Class [rdfs:Class]
Label:	Rights Statement
Definition:	A statement about the intellectual property rights (IPR) held in or over a Resource, a legal document giving official permission to do something with a resource, or a statement about access rights.

Term Name: Size	
URI:	http://purl.org/dc/terms/Size [dcterms:Size]
Type of Term:	http://www.w3.org/2000/01/rdf-schema#Class [rdfs:Class]
Definition:	The size of a digital resource or physical resource.
Comment:	For example, a size might be a number of pages or a specification of length, width and breadth.
Sub-Class Of:	http://purl.org/dc/terms/SizeOrDuration [dcterms:SizeOrDuration]

Term Name: SizeOrDuration	
URI:	http://purl.org/dc/terms/SizeOrDuration [dcterms:SizeOrDuration]
Type of Term:	http://www.w3.org/2000/01/rdf-schema#Class [rdfs:Class]
Label:	Size or Duration
Definition:	A size or duration.
Sub-Class Of:	http://purl.org/dc/terms/MediaTypeOrExtent [dcterms:MediaTypeOrExtent]

Term Name: Standard	
URI:	http://purl.org/dc/terms/Standard [dcterms:Standard]
Type of Term:	http://www.w3.org/2000/01/rdf-schema#Class [rdfs:Class]
Label:	Standard

2007-08-25 Usage Board meeting, Singapore Definition:	A basis for comparison; a reference point against which other things can be evaluated.

3. Domain and Range assertions for DCMI Properties

This section summarizes the assertions to be made for *has domain* and *has range* relationships for existing DCMI properties.

3.1. Properties of the Dublin Core Metadata Element Set, Version 1.1.

The terms described in this section are **existing** *properties* that have already been assigned URIs owned by DCMI, and for which term declarations are already made available by DCMI. **No changes** to those term declarations are required or proposed. No assertions of *has domain* or *has range* relationships will be made for the properties of the Dublin Core Metadata Element Set, Version 1.1. (This is equivalent to saying that for each of these properties there exists a *has domain* relationship with the class rdfs:Resource and a *has range* relationship with the class rdfs:Resource.)

Term Name: contributor	
URI:	http://purl.org/dc/elements/1.1/contributor [dc:contributor]
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]

Term Name: coverage	
URI:	http://purl.org/dc/elements/1.1/coverage [dc:coverage]
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]

Term Name: creator	
URI:	http://purl.org/dc/elements/1.1/creator [dc:creator]
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]

Term Name: date	
URI:	http://purl.org/dc/elements/1.1/date [dc:date]
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]

Term Name: description

URI:	http://purl.org/dc/elements/1.1/description [dc:description]
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]

Term Name: format	
URI:	http://purl.org/dc/elements/1.1/format [dc:format]
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]

Term Name: identifier	
URI:	http://purl.org/dc/elements/1.1/identifier [dc:identifier]
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]

Term Name: language	
URI:	http://purl.org/dc/elements/1.1/language [dc:language]
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]

Term Name: publisher	
URI:	http://purl.org/dc/elements/1.1/publisher [dc:publisher]
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]

Term Name: relation	
URI:	http://purl.org/dc/elements/1.1/relation [dc:relation]
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]

Term Name: rights	
URI:	http://purl.org/dc/elements/1.1/rights [dc:rights]

Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]

Term Name: source	
URI:	http://purl.org/dc/elements/1.1/source [dc:source]
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]

Term Name: subject	
URI:	http://purl.org/dc/elements/1.1/subject [dc:subject]
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]

Term Name: title	
URI:	http://purl.org/dc/elements/1.1/title [dc:title]
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]

Term Name: type	
URI:	http://purl.org/dc/elements/1.1/type [dc:type]
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]

3.2. Domain and Range assertions for the proposed new Properties of the Dublin Core Terms.

The terms described in this section are **new** *properties* that are to be assigned URIs owned by DCMI. The URIs will be in the *DCMI Namespace* http://purl.org/dc/terms. A **new** set of term declarations will be made available, and those **new** term declarations are summarized in this section. Those term declarations include:

- assertions of *sub-property of* relationships with other existing *properties*, and
- assertions of *has domain* and/or *has range* relationships with one or more of the *classes* specified above

The human-readable label, definition, comment etc for these new *properties* will be the same as for the corresponding existing *property* of the Dublin Core Metadata Element Set, Version 1.1.

Note: for completeness, the tables below include *has domain* and/or *has range* relationships with the class rdfs:Resource (italicised in the tables). Such a relationship is implicit even in the absence of an explicit *has domain* or *has range* assertion, so those assertions will not be present in the term

declarations. 2007-08-25 Usage Board meeting, Singapore

Term Name: contributor	
URI:	http://purl.org/dc/terms/contributor [dcterms:contributor]
Definition:	An entity responsible for making contributions to the resource.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/contributor [dc:contributor]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource* *[rdfs:Resource]*
Has Range:	http://purl.org/dc/terms/Agent [dcterms:Agent]

Term Name: coverage	
URI:	http://purl.org/dc/terms/coverage [dcterms:coverage]
Definition:	The spatial or temporal topic of the resource, the spatial applicability of the resource, or the jurisdiction under which the resource is relevant.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/coverage [dc:coverage]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource* *[rdfs:Resource]*
Has Range:	http://purl.org/dc/terms/LocationPeriodOrJurisdiction [dcterms:LocationPeriodOrJurisdiction]

Term Name: creator	
URI:	http://purl.org/dc/terms/creator [dcterms:creator]
Definition:	An entity primarily responsible for making the resource.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/terms/contributor [dcterms:contributor]

Sub-Property Of:	http://purl.org/dc/elements/1.1/creator [dc:creator]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource* *[rdfs:Resource]*
Has Range:	http://purl.org/dc/terms/Agent [dcterms:Agent]

Term Name: date	
URI:	http://purl.org/dc/terms/date [dcterms:date]
Definition:	A point or period of time associated with an event in the lifecycle of the resource.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/date [dc:date]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource* *[rdfs:Resource]*
Has Range:	http://www.w3.org/2000/01/rdf-schema#Literal [rdfs:Literal]

Term Name: description	
URI:	http://purl.org/dc/terms/description [dcterms:description]
Definition:	An account of the resource.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/description [dc:description]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource* *[rdfs:Resource]*
Has Range:	*http://www.w3.org/2000/01/rdf-schema#Resource* *[rdfs:Resource]*

Term Name: format	
URI:	http://purl.org/dc/terms/format [dcterms:format]
Definition:	The file format, physical medium, or dimensions of the resource.

Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/format [dc:format]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*
Has Range:	http://purl.org/dc/terms/MediaTypeOrExtent [dcterms:MediaTypeOrExtent]

Term Name: identifier	
URI:	http://purl.org/dc/terms/identifier [dcterms:identifier]
Definition:	An unambiguous reference to the resource within a given context.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/identifier [dc:identifier]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*
Has Range:	http://www.w3.org/2000/01/rdf-schema#Literal [rdfs:Literal]

Term Name: language	
URI:	http://purl.org/dc/terms/language [dcterms:language]
Definition:	A language of the resource.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/language [dc:language]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*
Has Range:	http://purl.org/dc/terms/LinguisticSystem [dcterms:LinguisticSystem]

Term Name: publisher	
URI:	http://purl.org/dc/terms/publisher [dcterms:publisher]

2007-08-25 Usage Board meeting, Singapore	
Definition:	An entity responsible for making the resource available.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/publisher [dc:publisher]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*
Has Range:	http://purl.org/dc/terms/Agent [dcterms:Agent]

Term Name: relation	
URI:	http://purl.org/dc/terms/relation [dcterms:relation]
Definition:	A related resource.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/relation [dc:relation]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*
Has Range:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*

Term Name: rights	
URI:	http://purl.org/dc/terms/rights [dcterms:rights]
Definition:	Information about rights held in and over the resource.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/rights [dc:rights]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*
Has Range:	http://purl.org/dc/terms/RightsStatement [dcterms:RightsStatement]

Term Name: source

URI:	http://purl.org/dc/terms/source [dcterms:source]
Definition:	A related resource from which the described resource is derived.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/terms/relation [dcterms:relation]
Sub-Property Of:	http://purl.org/dc/elements/1.1/source [dc:source]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource* *[rdfs:Resource]*
Has Range:	*http://www.w3.org/2000/01/rdf-schema#Resource* *[rdfs:Resource]*

Term Name: subject	
URI:	http://purl.org/dc/terms/subject [dcterms:subject]
Definition:	The topic of the resource.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/subject [dc:subject]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource* *[rdfs:Resource]*
Has Range:	*http://www.w3.org/2000/01/rdf-schema#Resource* *[rdfs:Resource]*

Term Name: title	
URI:	http://purl.org/dc/terms/title [dcterms:title]
Definition:	A name given to the resource.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/title [dc:title]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource* *[rdfs:Resource]*

Has Range:	http://www.w3.org/2000/01/rdf-schema#Literal [rdfs:Literal]

Term Name: type	
URI:	http://purl.org/dc/terms/type [dcterms:type]
Definition:	The nature or genre of the resource.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/type [dc:type]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource* *[rdfs:Resource]*
Has Range:	http://www.w3.org/2000/01/rdf-schema#Class [rdfs:Class]

3.3. Domain and Range assertions for the existing Properties of the Dublin Core Terms.

The terms described in this section are **existing** *properties* that have already been assigned URIs owned by DCMI, and for which term declarations are already made available by DCMI. Some **changes** to those existing term declarations are proposed, namely the addition of:

- assertions of *sub-property of* relationships with the new *properties* specified above, and
- assertions of *has domain* and/or *has range* relationships with one or more of the *classes* specified above

Note: for completeness, the tables below include *has domain* and/or *has range* relationships with the class rdfs:Resource (italicised in the tables). Such a relationship is implicit even in the absence of an explicit *has domain* or *has range* assertion, so those assertions will not be present in the term declarations.

Note that the term declarations listed here also reflect the changes proposed in the document *Revisions to DCMI Metadata Terms* [REVTERM].

Term Name: abstract	
URI:	http://purl.org/dc/terms/abstract [dcterms:abstract]
Definition:	A summary of the resource.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/description [dc:description]

2007-08-25 Sub-Property Of:	Usage Board meeting, Singapore http://purl.org/dc/terms/description [dcterms:description]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource* *[rdfs:Resource]*
Has Range:	*http://www.w3.org/2000/01/rdf-schema#Resource* *[rdfs:Resource]*

Term Name: accessRights	
URI:	http://purl.org/dc/terms/accessRights [dcterms:accessRights]
Definition:	Information about who can access the resource or an indication of its security status.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/rights [dc:rights]
Sub-Property Of:	http://purl.org/dc/terms/rights [dcterms:rights]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource* *[rdfs:Resource]*
Has Range:	http://purl.org/dc/terms/RightsStatement [dcterms:RightsStatement]

Term Name: accrualMethod	
URI:	http://purl.org/dc/terms/accrualMethod [dcterms:accrualMethod]
Definition:	The method by which items are added to a collection.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Has Domain:	http://purl.org/dc/dcmitype/Collection [dcmitype:Collection]
Has Range:	http://purl.org/dc/terms/MethodOfAccrual [dcterms:MethodOfAccrual]

Term Name: accrualPeriodicity	
URI:	http://purl.org/dc/terms/accrualPeriodicity [dcterms:accrualPeriodicity]
Definition:	The frequency with which items are added to a collection.

Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Has Domain:	http://purl.org/dc/dcmitype/Collection [dcmitype:Collection]
Has Range:	http://purl.org/dc/terms/Frequency [dcterms:Frequency]

Term Name: accrualPolicy	
URI:	http://purl.org/dc/terms/accrualPolicy [dcterms:accrualPolicy]
Definition:	The policy governing the addition of items to a collection.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Has Domain:	http://purl.org/dc/dcmitype/Collection [dcmitype:Collection]
Has Range:	http://purl.org/dc/terms/Policy [dcterms:Policy]

Term Name: alternative	
URI:	http://purl.org/dc/terms/alternative [dcterms:alternative]
Definition:	An alternative name for the resource.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/title [dc:title]
Sub-Property Of:	http://purl.org/dc/terms/title [dcterms:title]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*
Has Range:	http://www.w3.org/2000/01/rdf-schema#Literal [rdfs:Literal]

Term Name: audience	
URI:	http://purl.org/dc/terms/audience [dcterms:audience]
Definition:	A class of entity for whom the resource is intended or useful.

Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*
Has Range:	http://purl.org/dc/terms/AgentClass [dcterms:AgentClass]

Term Name: available	
URI:	http://purl.org/dc/terms/available [dcterms:available]
Definition:	Date (often a range) that the resource became or will become available.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/date [dc:date]
Sub-Property Of:	http://purl.org/dc/terms/date [dcterms:date]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*
Has Range:	http://www.w3.org/2000/01/rdf-schema#Literal [rdfs:Literal]

Term Name: bibliographicCitation	
URI:	http://purl.org/dc/terms/bibliographicCitation [dcterms:bibliographicCitation]
Definition:	A bibliographic reference for the resource.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/identifier [dc:identifier]
Sub-Property Of:	http://purl.org/dc/terms/identifier [dcterms:identifier]
Has Domain:	http://purl.org/dc/terms/BibliographicResource [dcterms:BibliographicResource]
Has Range:	http://www.w3.org/2000/01/rdf-schema#Literal [rdfs:Literal]

Term Name: conformsTo

URI:	http://purl.org/dc/terms/conformsTo [dcterms:conformsTo]
Definition:	An established standard to which the resource conforms.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/relation [dc:relation]
Sub-Property Of:	http://purl.org/dc/terms/relation [dcterms:relation]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource* *[rdfs:Resource]*
Has Range:	http://purl.org/dc/terms/Standard [dcterms:Standard]

Term Name: created	
URI:	http://purl.org/dc/terms/created [dcterms:created]
Definition:	Date of creation of the resource.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/date [dc:date]
Sub-Property Of:	http://purl.org/dc/terms/date [dcterms:date]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource* *[rdfs:Resource]*
Has Range:	http://www.w3.org/2000/01/rdf-schema#Literal [rdfs:Literal]

Term Name: dateAccepted	
URI:	http://purl.org/dc/terms/dateAccepted [dcterms:dateAccepted]
Definition:	Date of acceptance of the resource.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/date [dc:date]

Sub-Property Of:	http://purl.org/dc/terms/date [dcterms:date]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource* *[rdfs:Resource]*
Has Range:	http://www.w3.org/2000/01/rdf-schema#Literal [rdfs:Literal]

Term Name: dateCopyrighted	
URI:	http://purl.org/dc/terms/dateCopyrighted [dcterms:dateCopyrighted]
Definition:	Date of copyright.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/date [dc:date]
Sub-Property Of:	http://purl.org/dc/terms/date [dcterms:date]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource* *[rdfs:Resource]*
Has Range:	http://www.w3.org/2000/01/rdf-schema#Literal [rdfs:Literal]

Term Name: dateSubmitted	
URI:	http://purl.org/dc/terms/dateSubmitted [dcterms:dateSubmitted]
Definition:	Date of submission of the resource.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/date [dc:date]
Sub-Property Of:	http://purl.org/dc/terms/date [dcterms:date]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource* *[rdfs:Resource]*
Has Range:	http://www.w3.org/2000/01/rdf-schema#Literal [rdfs:Literal]

Term Name: educationLevel

URI:	http://purl.org/dc/terms/educationLevel [dcterms:educationLevel]
Definition:	A class of entity, defined in terms of progression through an educational or training context, for whom the resource is intended.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/terms/audience [dcterms:audience]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource* *[rdfs:Resource]*
Has Range:	http://purl.org/dc/terms/AgentClass [dcterms:AgentClass]

Term Name: extent	
URI:	http://purl.org/dc/terms/extent [dcterms:extent]
Definition:	The size or duration of the resource.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/format [dc:format]
Sub-Property Of:	http://purl.org/dc/terms/format [dcterms:format]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource* *[rdfs:Resource]*
Has Range:	http://purl.org/dc/terms/SizeOrDuration [dcterms:SizeOrDuration]

Term Name: hasFormat	
URI:	http://purl.org/dc/terms/hasFormat [dcterms:hasFormat]
Definition:	A related resource that is substantially the same as the pre-existing described resource, but in another format.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/relation [dc:relation]

Sub-Property Of:	http://purl.org/dc/terms/relation [dcterms:relation]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*
Has Range:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*

Term Name: hasPart	
URI:	http://purl.org/dc/terms/hasPart [dcterms:hasPart]
Definition:	A related resource that is included either physically or logically in the described resource.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/relation [dc:relation]
Sub-Property Of:	http://purl.org/dc/terms/relation [dcterms:relation]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*
Has Range:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*

Term Name: hasVersion	
URI:	http://purl.org/dc/terms/hasVersion [dcterms:hasVersion]
Definition:	A related resource that is a version, edition, or adaptation of the described resource.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/relation [dc:relation]
Sub-Property Of:	http://purl.org/dc/terms/relation [dcterms:relation]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*
Has Range:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*

2007-08-25 Usage Board meeting, Singapore **Term Name: instructionalMethod**	
URI:	http://purl.org/dc/terms/instructionalMethod [dcterms:instructionalMethod]
Definition:	A process, used to engender knowledge, attitudes and skills, that the described resource is designed to support.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource* *[rdfs:Resource]*
Has Range:	http://purl.org/dc/terms/MethodOfInstruction [dcterms:MethodOfInstruction]

Term Name: isFormatOf	
URI:	http://purl.org/dc/terms/isFormatOf [dcterms:isFormatOf]
Definition:	A related resource that is substantially the same as the described resource, but in another format.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/relation [dc:relation]
Sub-Property Of:	http://purl.org/dc/terms/relation [dcterms:relation]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource* *[rdfs:Resource]*
Has Range:	*http://www.w3.org/2000/01/rdf-schema#Resource* *[rdfs:Resource]*

Term Name: isPartOf	
URI:	http://purl.org/dc/terms/isPartOf [dcterms:isPartOf]
Definition:	A related resource in which the described resource is physically or logically included.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/relation [dc:relation]

Sub-Property Of:	http://purl.org/dc/terms/relation [dcterms:relation]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*
Has Range:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*

Term Name: isReferencedBy	
URI:	http://purl.org/dc/terms/isReferencedBy [dcterms:isReferencedBy]
Definition:	A related resource that references, cites, or otherwise points to the described resource.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/relation [dc:relation]
Sub-Property Of:	http://purl.org/dc/terms/relation [dcterms:relation]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*
Has Range:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*

Term Name: isReplacedBy	
URI:	http://purl.org/dc/terms/isReplacedBy [dcterms:isReplacedBy]
Definition:	A related resource that supplants, displaces, or supersedes the described resource.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/relation [dc:relation]
Sub-Property Of:	http://purl.org/dc/terms/relation [dcterms:relation]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*
Has Range:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*

2007-08-25 Usage Board meeting, Singapore **Term Name: isRequiredBy**	
URI:	http://purl.org/dc/terms/isRequiredBy [dcterms:isRequiredBy]
Definition:	A related resource that requires the described resource to support its function, delivery, or coherence.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/relation [dc:relation]
Sub-Property Of:	http://purl.org/dc/terms/relation [dcterms:relation]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource* *[rdfs:Resource]*
Has Range:	*http://www.w3.org/2000/01/rdf-schema#Resource* *[rdfs:Resource]*

Term Name: issued	
URI:	http://purl.org/dc/terms/issued [dcterms:issued]
Definition:	Date of formal issuance (e.g., publication) of the resource.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/date [dc:date]
Sub-Property Of:	http://purl.org/dc/terms/date [dcterms:date]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource* *[rdfs:Resource]*
Has Range:	http://www.w3.org/2000/01/rdf-schema#Literal [rdfs:Literal]

Term Name: isVersionOf	
URI:	http://purl.org/dc/terms/isVersionOf [dcterms:isVersionOf]
Definition:	A related resource of which the described resource is a version, edition, or adaptation.

2007-08-25 Usage Board meeting, Singapore	
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/relation [dc:relation]
Sub-Property Of:	http://purl.org/dc/terms/relation [dcterms:relation]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*
Has Range:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*

Term Name: license	
URI:	http://purl.org/dc/terms/license [dcterms:license]
Definition:	A legal document giving official permission to do something with the resource.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/rights [dc:rights]
Sub-Property Of:	http://purl.org/dc/terms/rights [dcterms:rights]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*
Has Range:	http://purl.org/dc/terms/LicenseDocument [dcterms:LicenseDocument]

Term Name: mediator	
URI:	http://purl.org/dc/terms/mediator [dcterms:mediator]
Definition:	A class of entity that mediates access to the resource and for whom the resource is intended or useful.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/terms/audience [dcterms:audience]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*

Has Range:	http://purl.org/dc/terms/AgentClass [dcterms:AgentClass]

2007-08-25 Usage Board meeting, Singapore

Term Name: medium	
URI:	http://purl.org/dc/terms/medium [dcterms:medium]
Definition:	The material or physical carrier of the resource.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/format [dc:format]
Sub-Property Of:	http://purl.org/dc/terms/format [dcterms:format]
Has Domain:	http://purl.org/dc/terms/PhysicalResource [dcterms:PhysicalResource]
Has Range:	http://purl.org/dc/terms/PhysicalMedium [dcterms:PhysicalMedium]

Term Name: modified	
URI:	http://purl.org/dc/terms/modified [dcterms:modified]
Definition:	Date on which the resource was changed.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/date [dc:date]
Sub-Property Of:	http://purl.org/dc/terms/date [dcterms:date]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*
Has Range:	http://www.w3.org/2000/01/rdf-schema#Literal [rdfs:Literal]

Term Name: provenance	
URI:	http://purl.org/dc/terms/provenance [dcterms: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.

Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*
Has Range:	http://purl.org/dc/terms/ProvenanceStatement [dcterms:ProvenanceStatement]

Term Name: references	
URI:	http://purl.org/dc/terms/references [dcterms:references]
Definition:	A related resource that is referenced, cited, or otherwise pointed to by the described resource.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/relation [dc:relation]
Sub-Property Of:	http://purl.org/dc/terms/relation [dcterms:relation]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*
Has Range:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*

Term Name: replaces	
URI:	http://purl.org/dc/terms/replaces [dcterms:replaces]
Definition:	A related resource that is supplanted, displaced, or superseded by the described resource.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/relation [dc:relation]
Sub-Property Of:	http://purl.org/dc/terms/relation [dcterms:relation]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*
Has Range:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*

2007-08-25 Usage Board meeting, Singapore

Term Name: requires	
URI:	http://purl.org/dc/terms/requires [dcterms:requires]
Definition:	A related resource that is required by the described resource to support its function, delivery, or coherence.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/relation [dc:relation]
Sub-Property Of:	http://purl.org/dc/terms/relation [dcterms:relation]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource* *[rdfs:Resource]*
Has Range:	*http://www.w3.org/2000/01/rdf-schema#Resource* *[rdfs:Resource]*

Term Name: rightsHolder	
URI:	http://purl.org/dc/terms/rightsHolder [dcterms:rightsHolder]
Definition:	A person or organization owning or managing rights over the resource.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource* *[rdfs:Resource]*
Has Range:	http://purl.org/dc/terms/Agent [dcterms:Agent]

Term Name: spatial	
URI:	http://purl.org/dc/terms/spatial [dcterms:spatial]
Definition:	Spatial characteristics of the resource.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/coverage [dc:coverage]
Sub-Property Of:	http://purl.org/dc/terms/coverage [dcterms:coverage]

Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*
Has Range:	http://purl.org/dc/terms/Location [dcterms:Location]

Term Name: tableOfContents	
URI:	http://purl.org/dc/terms/tableOfContents [dcterms:tableOfContents]
Definition:	A list of subunits of the resource.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/description [dc:description]
Sub-Property Of:	http://purl.org/dc/terms/description [dcterms:description]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*
Has Range:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*

Term Name: temporal	
URI:	http://purl.org/dc/terms/temporal [dcterms:temporal]
Definition:	Temporal characteristics of the resource.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/coverage [dc:coverage]
Sub-Property Of:	http://purl.org/dc/terms/coverage [dcterms:coverage]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*
Has Range:	http://purl.org/dc/terms/PeriodOfTime [dcterms:PeriodOfTime]

Term Name: valid	
URI:	http://purl.org/dc/terms/valid [dcterms:valid]

2007-08-25 Definition:	Usage Board meeting, Singapore Date (often a range) of validity of a resource.
Type of Term:	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property [rdf:Property]
Sub-Property Of:	http://purl.org/dc/elements/1.1/date [dc:date]
Sub-Property Of:	http://purl.org/dc/terms/date [dcterms:date]
Has Domain:	*http://www.w3.org/2000/01/rdf-schema#Resource [rdfs:Resource]*
Has Range:	http://www.w3.org/2000/01/rdf-schema#Literal [rdfs:Literal]

References

[DCAM]
Powell, Andy, Mikael Nilsson, Ambjörn Naeve, Pete Johnston and Thomas Baker. DCMI Abstract Model. DCMI Recommendation. June 2007.
<http://dublincore.org/documents/2007/06/04/abstract-model/>

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

[RDF]
Klyne, Graham and Jeremy Carroll, editors. Resource Description Framework: Concepts and Abstract Syntax. W3C Recommendation. 10 February 2004.
<http://www.w3.org/TR/rdf-concepts/>

[RDFS]
Brickley, Dan and R.V. Guha, editors. RDF Vocabulary Description Language 1.0: RDF Schema. W3C Recommendation. 10 February 2004.
<http://www.w3.org/TR/rdf-schema/>

[REVTERM]Baker, Thomas.
Revisions to DCMI Terms
2007-07-02<http://dublincore.org/usage/public-comment/documents/2007/07/dcterms-changes/>

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

2007-08-25 Usage Board meeting, Singapore

Issues Arising from Proposed Ranges for DCMI Properties 2007-07-02

This file: http://dublincore.org/usageboardwiki/RangesIssues

Refers to: *Domains and Ranges for DCMI Properties* 2007-07-02
http://dublincore.org/documents/2007/07/02/domain-range/

Introduction

The recent revision of the DCMI Abstract Model [DCAM] added the RDF Schema concepts of domain and range to the DCAM Vocabulary Model.

The motivation for assigning domains and ranges to DCMI properties is to make accessible to applications (through machine-processable descriptions of those properties) more of the information which is available to humans through the existing human-readable descriptions of those properties [DCRDFNOTES]. So for example, if the human-readable definition conveys (to a human reader) that, whenever a property is referenced in a statement, the value is an agent, the creation of a corresponding range assertion for the property enables an application to infer that, whenever it encounters a statement referencing that property, the value is an instance of the class of agents.

In the initial draft proposal for domains and ranges for DCMI properties circulated in February 2007 [DOMRANG0207], suggestions were made for domains and ranges for all current DCMI properties. At this point, it was not considered critical to specify explicitly that the classes specified as ranges were classes of literals or classes of non-literals. And for some cases at least, the proposal was that for a single property, values might be either literals in some statements or non-literals in others.

During the public comment period on this draft proposal, a new question was raised, that of compatibility with OWL-DL [OWL-OVER]. It was highlighted that it is problematic to allow a single property to take either literal values or non-literal values. As a result, during their meeting in March 2007, the DCMI Usage Board tried to decide for each DCMI property whether it should take literal values or non-literal values. As part of this exercise, it was proposed to create a new class of "non-literal resources" and to use that class as the range of some DCMI properties.

However, subsequent discussion suggests that it is not clear that simply assigning a range to a property of a class called, say, dcterms:NonLiteralResource would satisfactorily address the problem:

- Using only the constructs of RDF Schema included in the DCAM Vocabulary Model, it is not possible to "say" in machine-processable form that the class dcterms:NonLiteralResource excludes literals; that would require some additional constructs provided by OWL.
- We need to be careful to distinguish between supporting inferencing (which is what rdfs:range provides) and checking for consistency (which requires some other mechanism, such as that which will be provided by the Description Set Profile work currently in progress [DC-DSP]
- Fully addressing the question of compatibility with OWL-DL will require the use of the property typing provided by OWL rather than simply the use of rdfs:range

In short, while the exercise of analysing the DCMI properties to decide whether they should be used with literal or non-literal values was a useful and indeed necessary one, it is not clear that seeking to reflect those conclusions through the use of rdfs:range is appropriate.

Property: dcterms:title Proposed Range: rdfs:Literal

The proposal to assign a range for the property dcterms:title of the class rdfs:Literal means that when that property is referenced in a statement in a DC metadata description, the value is a literal, either plain or typed, and statements referencing this property should contain a literal value surrogate i.e. they should have the form (using the DC-Text syntax [DC-TEXT]):

```
@prefix dcterms: <http://purl.org/dc/terms/> .

DescriptionSet (
 Description (
  ResourceURI ( <http://example.org/123> )
  Statement (
   PropertyURI ( dcterms:title )
   LiteralValueString ( "Learning Biology"
    Language ( "en" )
   )
  )
 )
)
```

which, following the proposed guidelines for representing DC metadata descriptions using RDF [DC-RDF], would map to the single RDF triple:

```
@prefix dcterms: <http://purl.org/dc/terms/> .
```

```
<http://example.org/123> dcterms:title "Learning Biology"@en .
```

If a described resource is associated with other literals as titles, those literals form part of **separate** property-value pairs, and must be represented as **separate** statements in a DC metadata description.

```
@prefix dcterms: <http://purl.org/dc/terms/> .

DescriptionSet (
 Description (
  ResourceURI ( <http://example.org/123> )
  Statement (
   PropertyURI ( dcterms:title )
   LiteralValueString ( "Learning Biology"
    Language ( "en" )
   )
  )
  Statement (
   PropertyURI ( dcterms:title )
   LiteralValueString ( "Aprender la Biologia"
    Language ( "es" )
   )
  )
 )
)
```

which would map to the two RDF triples:

```
@prefix dcterms: <http://purl.org/dc/terms/> .

<http://example.org/123> dcterms:title "Learning Biology"@en ;
                         dcterms:title "Aprender la Biologia"@es ;
```

No relationship between those two literal values is indicated.

This is the case even where the literals are distinct representations of the same title, as is common in cultures where parallel writing systems are used. For example, in Japanese, the same title represented using the Kana and Kanji representations would have to be represented as two statements:

```
@prefix dcterms: <http://purl.org/dc/terms/> .

DescriptionSet (
 Description (
  ResourceURI ( <http://example.org/123> )
  Statement (
   PropertyURI ( dcterms:title )
   LiteralValueString ( "[Kanji character string]" )
  )
  Statement (
   PropertyURI ( dcterms:title )
   LiteralValueString ( "[Kana character string]" )
   )
  )
 )
)
```

The other consequence of the literal-as-value approach is that the value can not be further described in its own right: it can not be the *described resource* in a DC metadata description.

Range as "Sequence of Words"?

An alternative approach would be to assign a range for dcterms:title of some class other than the class of literals. One option would be to say that values of the property are "sequences of words". In this case the two Kana and Kanji literals could be treated in a DC metadata description as two different representations of a single non-literal value:

```
@prefix dcterms: <http://purl.org/dc/terms/> .

DescriptionSet (
```

```
 Description (
  ResourceURI ( <http://example.org/123> )
  Statement (
   PropertyURI ( dcterms:title )
   ValueString ( "[Kanji character string]" )
   ValueString ( "[Kana character string]" )
  )
 )
)
```

which would map to the three RDF triples:

```
@prefix dcterms: <http://purl.org/dc/terms/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .

<http://example.org/123> dcterms:title _:x .
_:x rdf:value "[Kanji character string]" ;
    rdf:value "[Kana character string]" .
```

where the relationship between the two literals is made explicit, though it also results in a more complex graph.

With this range, however, the case of the English and Spanish titles would still require two separate statements, as they are different "sequences of words".

Range as "Abstract Title"?

A third option would be to assign a range of some class which allowed the **different** language literals to be treated as representations of the same "abstract title" in which case the English and Spanish titles could be treated in a DC metadata description as two different representations of a single non-literal value:

```
@prefix dcterms: <http://purl.org/dc/terms/> .

DescriptionSet (
 Description (
  ResourceURI ( <http://example.org/123> )
  Statement (
   PropertyURI ( dcterms:title )
   ValueString ( "Learning Biology"
    Language ( "en" )
   )
   ValueString ( "Aprender la Biologia"
    Language ( "es" )
   )
  )
 )
)
```

which would map to the three RDF triples:

```
@prefix dcterms: <http://purl.org/dc/terms/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .

<http://example.org/123> dcterms:title _:x .
_:x rdf:value "Learning Biology"@en ;
    rdf:value "Aprender la Biologia"@es .
```

(This would probably be roughly equivalent to the way the LangString datatype is used in the IEEE Learning Object Metadata standard.)

Which class to use for the range of the dcterms:property is a choice, determined essentially by what precisely are the intended semantics of the property and what requirements it is intended to meet.

(All the above considerations apply also to the property dcterms:alternative, a subproperty of dcterms:title.)

Property: dcterms:date Proposed Range: rdfs:Literal

The proposal to assign a range for the property dcterms:date of the class rdfs:Literal means that when that property is referenced in a statement in a DC metadata description, the value is a literal, either plain or typed, and statements referencing this property should contain a literal value surrogate i.e. they should have the form:

```
@prefix dcterms: <http://purl.org/dc/terms/> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

DescriptionSet (
 Description (
  ResourceURI ( <http://example.org/123> )
  Statement (
   PropertyURI ( dcterms:date )
   LiteralValueString ( "2007-07-02"
    SyntaxEncodingSchemeURI ( xsd:date )
   )
  )
 )
)
```

which would map to the single RDF triple:

```
@prefix dcterms: <http://purl.org/dc/terms/> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

<http://example.org/123> dcterms:date "2007-07-02"^^xsd:date .
```

One of the advantages of this approach is that, as shown here, instances of the builtin XML Schema datatypes can be used as values.

As above for the case of dcterms:title, if a described resource is associated with other literals as dates, those literals form part of **separate** property-value pairs, and must be represented as **separate** statements in a DC metadata description. No relationship between those two literal values is indicated.

This is the case even where the literals are alternate notations for the same date (though datatype aware applications may be able to establish equivalences of the values represnted by different typed literals).

The other consequence of the literal-as-value approach is that the value can not be further described in its own right: it can not be the *described resource* in a DC metadata description.

Range of dcterms:date and range of dcterms:temporal

With regard to this last point, it is probably important to highlight that the current proposal for the dcterms:temporal property is to assign as range the class dcterms: PeriodOfTime, a class of "non-literal resources". This means that a statement referencing the dcterms:temporal property will contain a non-literal value surrogate, possibly with multiple value strings, whereas a statement referencing the dcterms:date property will contain a literal value surrogate, with a single value string, mapping to two different RDF graph "patterns". As a result, the same resource could not be the value in the case of a statement referencing the property dcterms: date and the case of a statement referencing the property dcterms:temporal.

```
@prefix dcterms: <http://purl.org/dc/terms/> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

DescriptionSet (
 Description (
  ResourceURI ( <http://example.org/123> )
  Statement (
   PropertyURI ( dcterms:date )
   LiteralValueString ( "start=2007-07-01; end=2007-07-02;"
    SyntaxEncodingSchemeURI ( dcterms:Period )
   )
  )
  Statement (
   PropertyURI ( dcterms:temporal )
   ValueString ( "start=2007-07-01; end=2007-07-02;"
    SyntaxEncodingSchemeURI ( dcterms:Period )
   )
  )
 )
)
```

This description set would map to the RDF triples:

```
@prefix dcterms: <http://purl.org/dc/terms/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
```

```
<http://example.org/123> dcterms:date "start=2007-07-01; end=2007-07-02;"^^dcterms:Period ;
                         dcterms:temporal _:x .
_:x rdf:value "start=2007-07-01; end=2007-07-02;"^^dcterms:Period .
```

And, further, an application processing data containing statements referencing both properties will have to use these two **different** patterns when selecting/querying using the two different properties.

An alternative approach would be to assign a range for the dcterms:date property of some class of non-literal resources, e.g. the class of "dates or periods".

(All the above considerations apply also to all the subproperties of dcterms:date.)

Property: dcterms:description Proposed Range: rdfs:Resource

The proposal for the property dcterms:description is not to assign a range, which is the equivalent of saying that the range is the class rdfs:Resource. This means that when that property is referenced in a statement in a DC metadata description, the value may be a resource of any type, and statements referencing this property may contain a literal value surrogate or may contain a non-literal value surrogate i.e. they may have the form (literal value surrogate case):

```
@prefix dcterms: <http://purl.org/dc/terms/> .

DescriptionSet (
 Description (
  ResourceURI ( <http://example.org/123> )
  Statement (
   PropertyURI ( dcterms:description )
   LiteralValueString ( "The textbook Learning Biology provides an introduction to the topic."
    Language ( "en" )
   )
  )
 )
)
```

which would map to the RDF triple:

```
@prefix dcterms: <http://purl.org/dc/terms/> .

<http://example.org/123> dcterms:description "The textbook Learning Biology provides an introduction to the
topic."@en .
```

Or they may have the form (non-literal value surrogate case):

```
@prefix dcterms: <http://purl.org/dc/terms/> .

DescriptionSet (
 Description (
  ResourceURI ( <http://example.org/123> )
  Statement (
   PropertyURI ( dcterms:description )
   ValueURI ( <http://example.org/123/description> )
  )
 )
)
```

which would map to the RDF triple:

```
@prefix dcterms: <http://purl.org/dc/terms/> .

<http://example.org/123> dcterms:description <http://example.org/123/description> .
```

The choice to allow both literal and non-literal values reflects the definition of the dcterms:description property (which explicitly allows for values to be "graphical representations"), and for the existing use of the current dc:description property on which the new property is based, where values are often resources referenced by URIs.

OWL-DL compatibility

However, this use of the dcterms:description property with both literal and non-literal values does mean that this property and its subproperties (dcterms:abstract, dcterms:tableOfContents) would mean that it is problematic to use this property in OWL-DL. One possible solution to this would be to define two separate properties for use in that context, one an OWL datatypeProperty (with literal values) and the second an OWL objectProperty (with individuals as values).

References

[DCAM]
DCMI Abstract Model 2007-06-04
http://dublincore.org/documents/2007/06/04/abstract-model/

[DC-RDF-NOTES]
Notes on DCMI specifications for Dublin Core metadata in RDF 2007-04-02
http://dublincore.org/documents/2007/04/02/dc-rdf-notes/

[DOMRANG0207]
Domains and Ranges for DCMI Properties 2007-02-05
http://dublincore.org/documents/2007/02/05/domain-range/

[OWL-OVER]
OWL Web Ontology Language Overview W3C Recommendation 10 February 2004.
http://www.w3.org/TR/2004/REC-owl-features-20040210/

[DC-TEXT]
DC-Text: A Text Syntax for Dublin Core Metadata Draft of 2007-04-02.
http://dublincore.org/architecturewiki/DCText/2007-04-02

[DC-DSP]
DCMI Description Set Profile Working Draft.
http://dublincore.org/architecturewiki/DescriptionSetProfile

2007-06-29 DCMI & Non-Literal Resources - "Pete Johnston" <Pete.Johnston@eduserv.org.uk>

To respond to Alistair's question about what it is we're trying to achieve, and to try to explain how we've got to where we are, my understanding is as follows:

(i) The initial motivation for assigning domains/ranges - as stated by Tom in the announcement [1] - was to make explicit to applications reading the RDFS data what was already implicit to humans reading the human-readable comments (e.g. when I make a statement - any statement - using dcterms:creator, the value is an Agent).

This seems to be quite in line with Alistair's remindeer that we should "assign ranges to license inferencing". (There's probably an argument that there's not a great deal of value in allowing a consumer to infer that a value is an "AccrualMethod" without getting any more info about the relationships of that class to any other classes, but apart from that, I don't think there's a problem there.)

(ii) At this point we didn't care too much if we said that a property had a range of rdfs:Resource (or we left the range unspecified) and allowed for values to be either literals or non-literals (and we changed the DCAM description model so that for any individual statement it is unambiguous whether the value is a literal or not.)

Having said that, I think in some cases we really did intend to specify that a property should be used only with a "non-literal resource" as value (e.g. I think this is the case for dcterms:relation and its subproperties). _However_ taking Alistair's point about inferencing v enforcing consistency, then maybe the latter is an example of wanting to enforce consistency, and using rdfs:range is not the way to do that. (I guess the exception where rdfs:range could work in this way would be when an XML Schema datatype I used as a range? e.g. a datatype-aware application could detect that "xyz" wasn't in the lexical space of xsd:int.)

Also, defining a class called dcterms:NonLiteralResource and saying in human-readable terms "This is a class of everything except literals" and using that as a range doesn't in itself do much good: either an application needs some "built-in knowledge" about that class or we need to be able to say in machine-processable terms, "This class excludes literals" and we need OWL constructs to say that.

(Aside: Doesn't this apply for all the other classes too? i.e. how do I know the class dcterms:Agent excludes literals? If we say

dcterms:creator rdfs:range dcterms:Agent .

and I find

document:D dcterms:creator "Fred" .

Does an "RDFS-aware" application see a "contradiction" there? Probably not?

But if we say

dcterms:date rdfs:range rdfs:Literal .

and I find

```
document:D dcterms:date period:FirstWorldWar .
period:FirstWorldWar rdfs:label "First World War" .
period:FirstWorldWar some:start "1914"^^xsd:date .
period:FirstWorldWar some:end "1918"^^xsd:date .
```

then it would?)

(iii) Then in the course of the public comment period on the first proposal, the new question of OWL-DL compatability was raised. And it was suggested that allowing a single property to take either literal values in or non-literal values was problematic, at least from an OWL-DL perspective.

So at their last meeting, the DCMI Usage Board tried to decide for each property whether it should take literal values or non-literal values.

I think we're saying that while this is a worthwhile - and probably necessary - exercise, trying to express the results of that exercise using rdfs:range is not the right thing to do. For checking consistency in OWL-DL, we really need to type properties as owl:DatatypeProperty or owl:ObjectProperty (which is what I see the FOAF folk seem to have done), and that may even involve defining some "parallel" properties (e.g. description as object and description as literal).

So I think in the short term, I think we're saying we should

- for those cases where we were going to specify range = "Non-Literal Resource", leave them as range = rdfs:Resource (no range specified)
- note that we haven't addressed the OWL-DL compatability problem

[1] http://www.jiscmail.ac.uk/cgi-bin/webadmin?A2=ind0702&LUe-ARCHITECTURE&P=291

2007-07-15 From: Mikael Nilsson <mikael@nilsson.name>

fre 2007-06-29 klockan 12:40 +0100 skrev Pete Johnston:
> Having said that, I think in some cases we really did intend to specify
> that a property should be used only with a "non-literal resource" as
> value (e.g. I think this is the case for dcterms:relation and its
> subproperties). _However_ taking Alistair's point about inferencing v
> enforcing consistency, then maybe the latter is an example of wanting to
> enforce consistency, and using rdfs:range is not the way to do that. (I
> guess the exception where rdfs:range could work in this way would be
> when an XML Schema datatype I used as a range? e.g. a datatype-aware
> application could detect that "xyz" wasn't in the lexical space of
> xsd:int.)

Well, I must say I disagree with Alistairs view that rdfs:range and rdfs:domain are *only* used to enable inferencing. Even according to the RDF Primer [1]:

"RDF Schema also provides vocabulary for describing how properties and classes are intended to be used together in RDF data."

"The rdfs:domain property is used to indicate that a particular property applies to a designated class."

"This schema-supplied information might be used in different ways. One application might interpret this statement as specifying part of a template for RDF data it is creating, and use it to ensure that any ex:author property has a value of the indicated (ex:Person) class. That is, this application interprets the schema description as a constraint in the same way that a programming language might. However, another application might interpret this statement as providing additional

information about data it is receiving, information which may not be provided explicitly in the original data. For example, this second application might receive some RDF data that includes an ex:author property whose value is a resource of unspecified class, and use this schema-provided statement to conclude that the resource must be an instance of class ex:Person. A third application might receive some RDF data that includes an ex:author property whose value is a resource of class ex:Corporation, and use this schema information as the basis of a warning that "there may be an inconsistency here, but on the other hand there may not be". Somewhere else there may be a declaration that resolves the apparent inconsistency (e.g., a declaration to the effect that "a Corporation is a (legal) Person")."

Thus, ranges are indeed *intended* to convey an idea of how the property is *intended* to be used.

> Also, defining a class called dcterms:NonLiteralResource and saying in
> human-readable terms "This is a class of everything except literals" and
> using that as a range doesn't in itself do much good: either an
> application needs some "built-in knowledge" about that class or we need
> to be able to say in machine-processable terms, "This class excludes
> literals" and we need OWL constructs to say that.

That does not mean that it's useless. It just means that not all application will be able to use it as a consistency check. But the explicit *intention* has still been documented.

> (Aside: Doesn't this apply for all the other classes too? i.e. how do I
> know the class dcterms:Agent excludes literals? If we say
>
> dcterms:creator rdfs:range dcterms:Agent .
>
> and I find
>
> document:D dcterms:creator "Fred" .
>
> Does an "RDFS-aware" application see a "contradiction" there? Probably
> not?

It can conclude that dcterms:Agent rdfs:subClassOf rdfs:Literal, I suppose.

> But if we say
>
> dcterms:date rdfs:range rdfs:Literal .
>
> and I find
>
> document:D dcterms:date period:FirstWorldWar .
> period:FirstWorldWar rdfs:label "First World War" .
> period:FirstWorldWar some:start "1914"^^xsd:date .
> period:FirstWorldWar some:end "1918"^^xsd:date .
>
> then it would?)

No. It would just conclude

period:FirstWorldWar rdf;type rdfs:Literal

:-/

> I think we're saying that while this is a worthwhile - and probably
> necessary - exercise, trying to express the results of that exercise

> using rdfs:range is not the right thing to do. For checking consistency
> in OWL-DL, we really need to type properties as owl:DatatypeProperty or
> owl:ObjectProperty (which is what I see the FOAF folk seem to have
> done), and that may even involve defining some "parallel" properties
> (e.g. description as object and description as literal).

I don't see why we can't do

```
dcterms:NonLiteralResource a rdfs:Class ;
   rdf:comment "The class of resources which are not literals" ;
   owl:disjointWith rdfs:Literal .
```

Together with

```
dcterms:Agent rdf:type dcterms:NonLiteralResource.
```

and

```
document:D dcterms:creator "Fred" .
```

we would get

```
document:D dcterms:creator _:x .
_:x rdf:type rdfs:Literal
```

and
```
_:x rdf:type dcterms:Agent
```

and thus a contradiction.

What's wrong with that?

> So I think in the short term, I think we're saying we should
>
> - for those cases where we were going to specify range = "Non-Literal
> Resource", leave them as range = rdfs:Resource (no range specified)

I don't agree. Based on this discussion, I really feel we *should*
introduce the NonLiteral class.

> - note that we haven't addressed the OWL-DL compatability problem

As far as I understood it, this is more a case of ontologies needing to
declare the properties locally as ObjectProperty or DatatypeProperty. A
property cannot be both *in one ontology*. Right?

/Mikael

[1] http://www.w3.org/TR/rdf-primer/

2007-07-04
Date: Wed, 4 Jul 2007 07:05:12 -0400
From: Bruce D'Arcus <bdarcus@GMAIL.COM>
Subject: Re: Public Comment on domains and ranges - special cases
To: DC-ARCHITECTURE@JISCMAIL.AC.UK

Thomas Baker wrote:

> dcterms:description - rdfs:Resource

In Pete's summary, he says of this decision:

"However, this use of the dcterms:description property with both literal and non-literal values does mean that this property and its subproperties (dcterms:abstract, dcterms:tableOfContents) would mean that it is problematic to use this property in OWL-DL. One possible solution to this would be to define two separate properties for use in that context, one an OWL datatypeProperty (with literal values) and the second an OWL objectProperty (with individuals as values)."

It also just seems to me problematic from a basic query and processing standpoint irrespective of OWL.

So what do people think of two properties for description, then? Maybe dcterms:description and dcterms:richDescription or some such ...

2007-07-04
From: Pete Johnston <Pete.Johnston@EDUSERV.ORG.UK>
Subject: Re: Public Comment on domains and ranges - special cases
To: DC-ARCHITECTURE@JISCMAIL.AC.UK

Bruce said:

> In Pete's summary, he says of this decision:
>
> "However, this use of the dcterms:description property with
> both literal and non-literal values does mean that this
> property and its subproperties (dcterms:abstract,
> dcterms:tableOfContents) would mean that it is problematic to
> use this property in OWL-DL. One possible solution to this
> would be to define two separate properties for use in that
> context, one an OWL datatypeProperty (with literal values)
> and the second an OWL objectProperty (with individuals as values)."
>
> It also just seems to me problematic from a basic query and
> processing standpoint irrespective of OWL.

Ah, thanks. Good point. I was making pretty much that sort of that argument for the differences in the dc:date/dcterms:temporal case, without realising that of course it would apply to the dcterms:description case too. All the more so as it's a single property.

> So what do people think of two properties for description,
> then? Maybe dcterms:description and dcterms:richDescription
> or some such ...

The coining of the new dcterms properties at this point does give us an opportunity to nail this (and it would provide the basis for a

"smoother" forward path if/when we come to grasp the OWL-DL nettle).

We'd need to think about whether the subproperties needed "parallel forms" too. (I suspect both abstract and tableOfContents are being deployed to refer to separate documents as values, as well as to literal values).

2007-07-04
Date: Wed, 4 Jul 2007 08:24:11 -0400
Reply-To: DCMI Architecture Forum <DC-ARCHITECTURE@JISCMAIL.AC.UK>
Sender: DCMI Architecture Forum <DC-ARCHITECTURE@JISCMAIL.AC.UK>
From: Bruce D'Arcus <bdarcus@GMAIL.COM>
Subject: Re: Public Comment on domains and ranges - special cases
To: DC-ARCHITECTURE@JISCMAIL.AC.UK

On Jul 4, 2007, at 7:53 AM, Pete Johnston wrote:

>> So what do people think of two properties for description,
>> then? Maybe dcterms:description and dcterms:richDescription
>> or some such ...
>
> The coining of the new dcterms properties at this point does give us an
> opportunity to nail this (and it would provide the basis for a
> "smoother" forward path if/when we come to grasp the OWL-DL nettle).
>
> We'd need to think about whether the subproperties needed "parallel
> forms" too. (I suspect both abstract and tableOfContents are being
> deployed to refer to separate documents as values, as well as to
> literal
> values).

I'd tend to think of abstract as a literal, and a TOC as a resource.

BTW, the OASIS OpenDocument TC just approved our new RDF-based metadata proposal for the format. I'd like to suggest implementors use the new dcterms properties. Is it appropriate to do that now?

And if yes, where can I point them for basic information?

This document is sort of what I'm looking for, but is rather abstract.

<http://dublincore.org/documents/2007/06/04/dc-rdf/>

It's heavily focused on the DC abstract model and syntax. I want to point developers to some basic RDF documentation and to a page at the DCMI site that discuses how to encode the DC terms. I don't want to expect them to have to learn another -- largely overlapping but different -- model and syntax just to be able to do this.

This document is a little more RDF focused ...

<http://dublincore.org/documents/dc-rdf-notes/>

.. but is missing the examples. This document has examples, but they're dated (no dcterms:creator and such):

<http://dublincore.org/documents/dcmes-xml/>

Any suggestions? I'd almost like to see the last document updated for 2007.

BTW, I did some work on a RELAX NG schema for DC awhile back. I'd be
happy to donate it if there's any interest.

2007-07-30
Date: Fri, 27 Jul 2007 21:09:42 +0900
Reply-To: DCMI Architecture Forum <DC-ARCHITECTURE@JISCMAIL.AC.UK>
Sender: DCMI Architecture Forum <DC-ARCHITECTURE@JISCMAIL.AC.UK>
From: MIYAZAWA Akira <miyazawa@NII.AC.JP>
Subject: Domains and Ranges Public Comment : dc:title, multi-script and literal
To: DC-ARCHITECTURE@JISCMAIL.AC.UK

All,

This is a comment for dc:title as literal.
The range of dc:title should not be "literal" if it means
a "sequence of characters".
I would like to point that a literal with language identification
is not just a "sequence of characters". Multiple script
representation of title (or any text) in Japanese is an example.
Even in English, "DUBLIN CORE METADATA INITIATIVE" is same name
as "Dublin Core Metadata Initiative", though they are not identical
as a "sequence of characters".

One may argue that well established tradition of indexing and
software techniques can cover this case issues. Yes, but, it
is a hard-coding of language dependent (implicit) knowledge.

I think our goal is to describe the knowledge explicitly.

What most people think as a literal is, in most cases, a text
in some language. In this sense, dc:title may a literal.
But, it is no just a "sequence of characters".
(It may be a subclass of "sequence of characters", though.)

I know my argument is not well organized. But I was contemplating
this, since this spring and still in this stage...

Best wishes,

Akira MIYAZAWA
National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo 101-8430
Japan
Phone: +81-3-4212-2508
Fax: +81-3-3556-1916
email: miyazawa@nii.ac.jp

Title: Description Set Profile
Identifier: http://stage.dublincore.org/usage/meetings/2007/08/singapore/.html/description-set-profile.html
Created: 2007-07-19

Date: Sat, 14 Jul 2007 13:45:54 +0200
Reply-To: DCMI Architecture Forum <DC-ARCHITECTURE@JISCMAIL.AC.UK>
Sender: DCMI Architecture Forum <DC-ARCHITECTURE@JISCMAIL.AC.UK>
From: Mikael Nilsson <mikael@NILSSON.NAME>
Subject: DCMI Description Set Profiles - a basis for application profiles
To: DC-ARCHITECTURE@JISCMAIL.AC.UK

Hi everyone!

Over the last couple of months, there has been some brainstorming going on between me, Pete, Andy and Tom, regarding what it means to define a formal DCMI Application Profile Model.

In this mail, I'm going to quickly present the current proposal: the Description Set Profile Model. Note that a) this is a work in progress, still in the design phase, and b) a fuller presentation of the idea will be made at te DC2007 conference.

The basic idea is that at the core of an application profile, is a set of constraints on the way a metadata record may be constructed. Of course, an application profile in general can be much more, but assume we disregard all of the following potential aspects of application profiles:

* Definitions of vocabulary
* Vocabulary versioning/managamen
* Human-readable comments
* and so on

and just stick with the basic notion of a constraint language for records. We then end up with what we are currently calling a Description Set Profile. There have been a few other attempts at doing this, most with a single fundamental issue: they cannot describe a Description Set consisting of multiple descriptions. Given complex needs such as the ePrints AP, the RDA AP or the IEEE LOM AP, this is not acceptable.

I am in the process of fleshing out a description of such a DSP model, based on discussions between me, Pete, Andy and Tom and others. The draft I'm working on is available here:

http://dublincore.org/architecturewiki/DescriptionSetProfile

This is still a *very* rough and incomplete draft, but I wanted to get it out for comment as early as possible.

Some comments:

* The basic idea is to describe:

1. What kinds of descriptions might be in the description set, and their relations
2. What kind of statements that can occur in those descriptions
3. What kind of constructs that are accepted in the statements.

* The above constraints can then be used to, for example:

* Validate a given record against a DSP
* Automatically configure metadata editors for a given DSP
* Create database schemas or query interfaces for a given DSP
* etc.

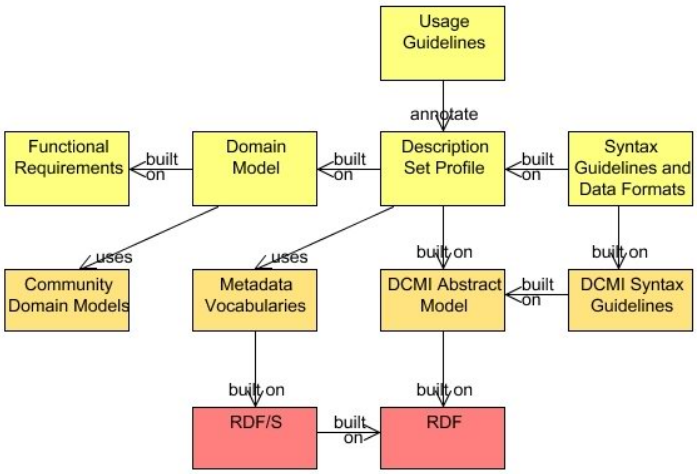
* A full-fledged Application Profile would the a DSP to describe the syntactic constraints, to which it may add a number of different things, such as functional requirements, domain models, usage guidelines, etc etc

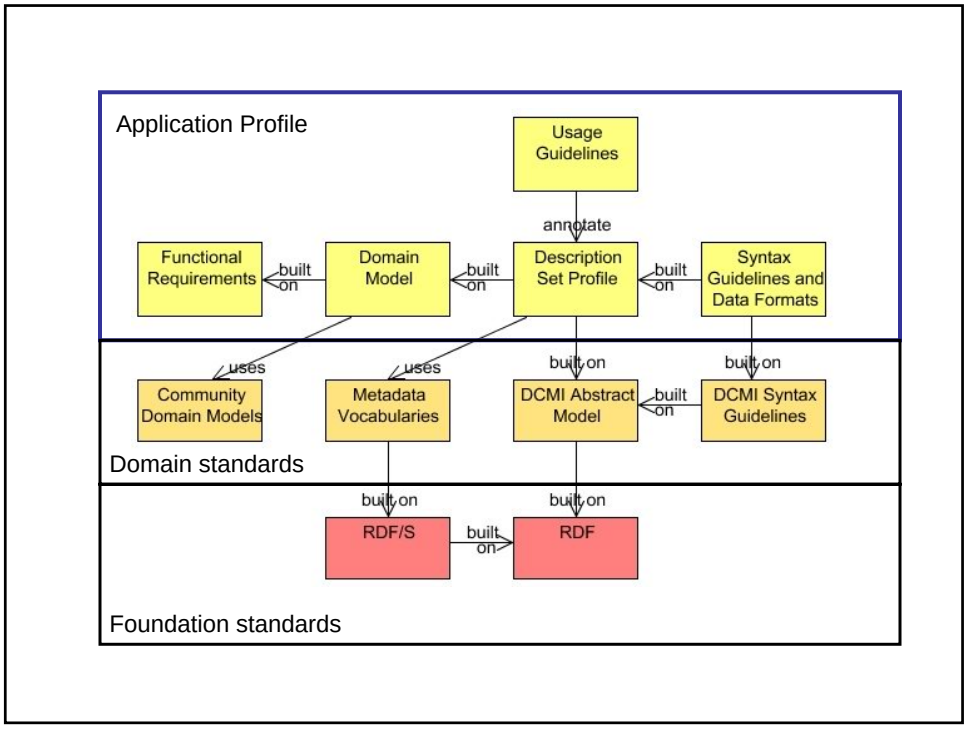
* Note that the proposed constraint language is relatively simplistic, and based on perceived needs of the Application Profiles currently in development. For more complex scenarios, you're probably better of using a syntax-specific method, such as XML Schema, XML Query, SPARQL or similar.

* On the other hand, the simplicity makes it conceivable that a DSP can be relatively easily converted into an XML Schema or an RDF graph pattern (along the lines of SHAME: http://kmr.nada.kth.se/shame).

What is a DC Application Profile?

- A DCAM-conformant Application Profile ("DC Application Profile") is packet of documentation which consists of:
  - Functional requirements (desirable)
  - Domain model (mandatory)
  - Description Set Profile (DSP) (mandatory)
  - Usage guidelines (optional)
  - Encoding syntax guidelines (optional)




Application Profile
Usage Guidelines
annotate
Functional Requirements
built on
Domain Model
built on
Description Set Profile
built on
Syntax Guidelines and Data Formats
uses
uses
built on
built on
Community Domain Models
Metadata Vocabularies
DCMI Abstract Model
built on
DCMI Syntax Guidelines
Domain standards
built on
built on
RDF/S
built on
RDF
Foundation standards

2007-08-25 Usage Board meeting, Singapore

DCMI Description Set Profile

Important: This document is currently in draft. Comments should please be sent to DC-ARCHITECTURE@JISCMAIL.AC.UK (Subscription and archives) .

Editor: Mikael Nilsson <mikael@nilsson.name>

1. TODO
2. Issues so far
3. Introduction
4. Basic structure
5. Basic semantics
6. Usage examples
    1. Example 1: Constraining the resource
    2. Example 2: Constraining a property
    3. Example 3: Constraining the value
    4. Example 4: Two resources
7. Description Templates
    1. Identifier
    2. Standalone
    3. Minimum occurrence constraint
    4. Maximum occurrence constraint
    5. Resource Class Membership Constraint
8. Statement templates
    1. Minimum occurrence constraint
    2. Maximum occurrence constraint
    3. Type constraint
    4. Property constraints
        1. Property list constraint
        2. Sub-property constraint
    5. Literal value constraints
        1. Literal list constraint
        2. Literal language constraint
        3. Literal language list constraint
        4. Syntax Encoding Scheme constraint
        5. Syntax Encoding Scheme list constraint
    6. Non-literal value constraints
        1. Description template reference
        2. Class membership constraint
        3. Value URI constraint
            1. Value URI occurrence constraint
            2. Value URI list constraint
        4. Vocabulary encoding scheme constraint
            1. Vocabulary encoding scheme occurrence constraint
            2. Vocabulary encoding scheme list constraint
        5. Value String Constraints
            1. Minimum occurrence constraint
            2. Maximum occurrence constraint
            3. Other constraints
9. XML structure
10. RDF variant
11. Examples
12. Simple FOAF

1. TODO

- Add hotlinks to DCAM concepts
- More examples
- Verify and test XSD

2. Issues so far

- Simultaneous subproperty and property list constraints?
- nonliteral constraint in the case type=unspecified.
- class constraint, relating descriptions and "standalone" descriptions
- description template matching based only on resource type, not path.

3. Introduction

The DCMI Description Set Profile specification describes an information model and XML expression of a "Description Set Profile" (DSP). The term *description set* and the associated concepts used in this specification are defined as in the DCMI Abstract model [DCAM].

A DSP is a way of describing structural constraints on a description set. It constrains the resources that may be described by descriptions in the description set, the properties that may be used, and the ways a value surrogate may be given.

A DSP can be used for many different purposes, such as:

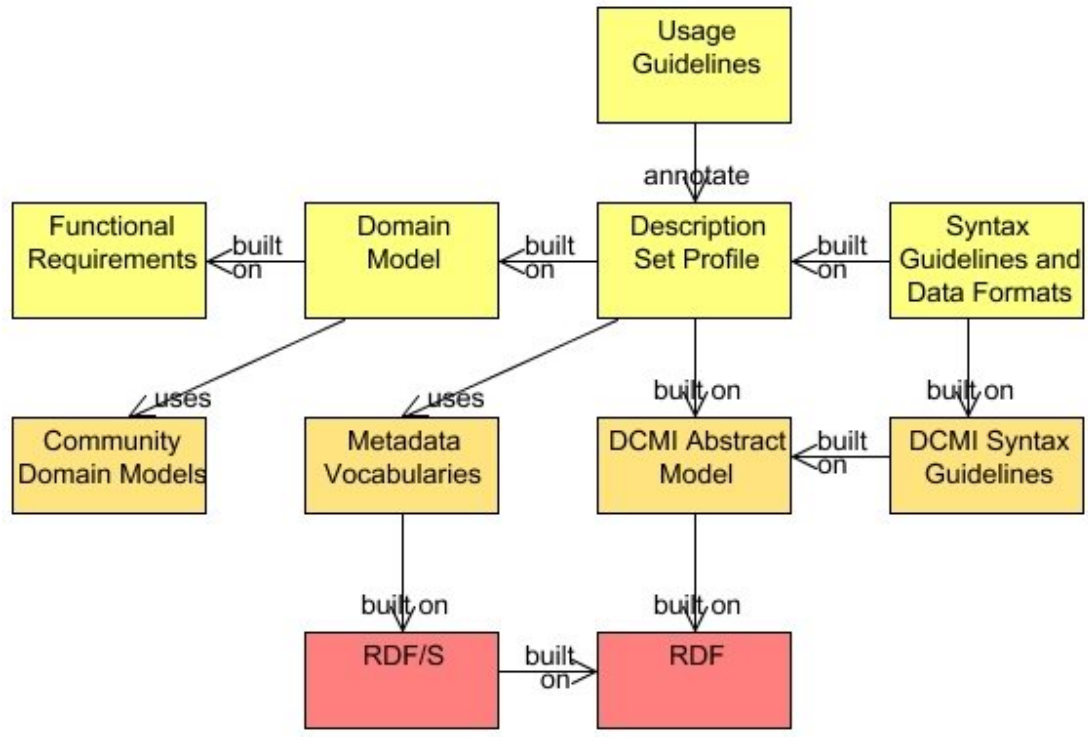
- as a formal representation of the constraints of a Dublin Core Application Profile
- as configuration for databases
- as configuration for metadata editing tools
- etc.

A DSP does *not* address the following:

- Human-readable documentation.
- Definition of vocabularies.
- Version control.

A DSP contains the formal syntactic constraints only, and will need to be combined with human-readable information, usage guidelines, version management, etc. in order to be used as an application profile. However, the design of the DSP information model is intended to facilitate the merging of DSP information and external information of the above kinds, for example by tools generating human-readable documentation for a Dublin Core Application Profile.

A Dublin Core Application Profile is a document, or set of documents, that puts a Description Set Profile into a broader context of Functional Requirements, Domain Models (such as Functional Requirements for Bibliographic Records), guidelines on syntax and usage, and possibly data formats. The following figure depicts the basic elements of a Dublin Core Application Profile (this model will be elaborated in future documents).



4. Basic structure

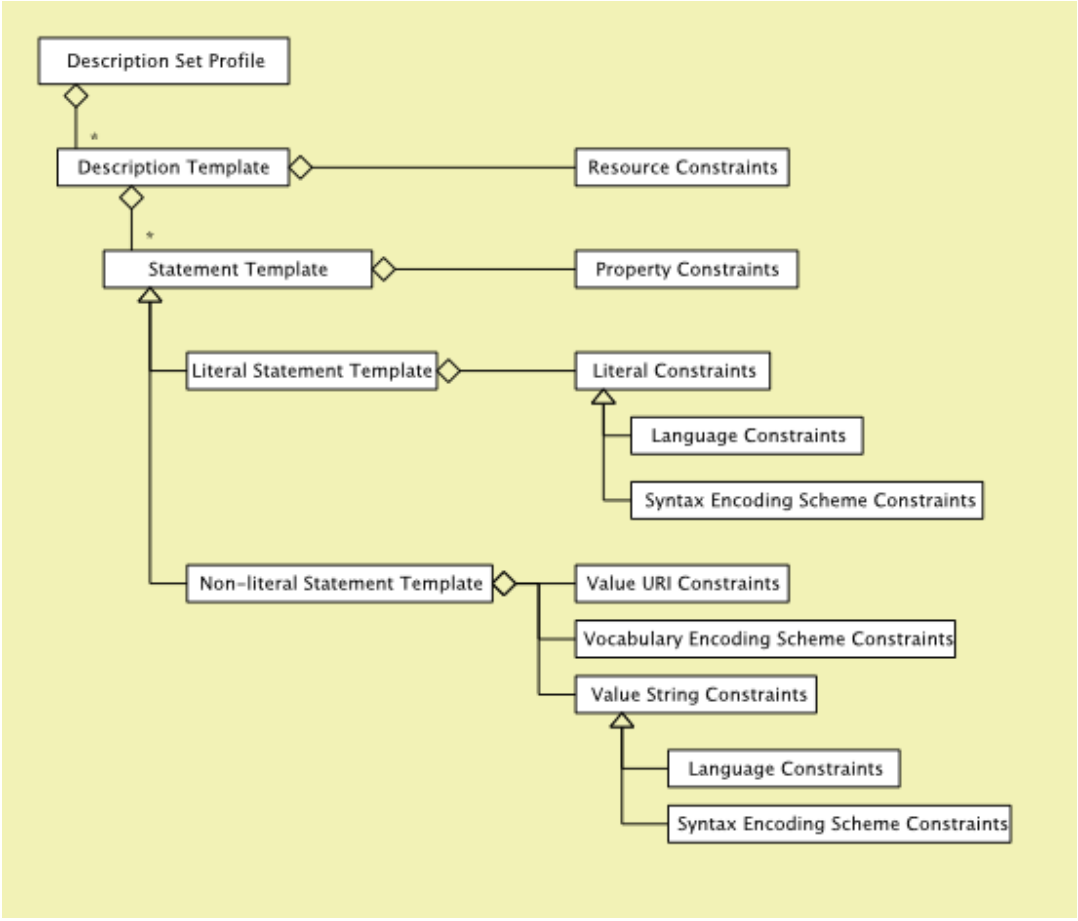
A DSP describes the structure of a Description Set by using the notions of "templates" and "constraints". A template describes the possible metadata structures in a conforming record.

There are two levels of templates in a Description Set Profile:

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

While templates are used to express structures, constraints are used to limit those structures. The following figure depicts the basic elements of the structure.

2007-08-25 Usage Board meeting, Singapore



5. Basic semantics

The fundamental usage model for a DSP is to examine whether a metadata record *matches* the DSP.

Matching of a description set is defined as follows:

Binding of descriptions to description templates

> Each description is bound to a Description Template by evaluating the *Resource Constraint* of each Description Template against the described resource. Each description must be bound to exactly one Description Template.

Binding of statements to statement templates

> For each description, each statement is bound to a Statement Templates in the corresponding Description Template by evaluating the *Property Constraint*. Each statement must be bound to exactly one Statement Template.

Evaluating constraints

> Now that all metadata in the description set has been bound to a template, all constraints can be verified.

6. Usage examples

6.1. Example 1: Constraining the resource

The following DSP matches descriptions with a single resource. The resource must be an instance of `foaf:Person`.

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

<DescriptionSetProfile xmlns:foaf="http://xmlns.com/foaf/0.1/">

  <DescriptionTemplate ID="person" minOccurs="1" maxOccurs="1" standalone="yes">

    <ResourceConstraint>
      <MemberOf>
        <Option value="foaf:Person"/>
      </MemberOf>
    </ResourceConstraint>
```

```
  </DescriptionTemplate>

</DescriptionSetProfile>
```

As it stands, this DSP does not allow for the description of that resource to contain any statements, so it is not very useful.

6.2. Example 2: Constraining a property

The following DSP adds a mandatory foaf:name property with a literal value to the previous example.

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

<DescriptionSetProfile xmlns:foaf="http://xmlns.com/foaf/0.1/">

  <DescriptionTemplate ID="person" minOccurs="1" maxOccurs="1" standalone="yes">

    <ResourceConstraint>
      <MemberOf>
        <Option value="foaf:Person"/>
      </MemberOf>
    </ResourceConstraint>

    <StatementTemplate minOccurs="1" maxOccurs="1" type="literal">
      <PropertyConstraint>
        <InSet>
          <Option value="foaf:name"/>
        </InSet>
      </PropertyConstraint>
    </StatementTemplate>

  </DescriptionTemplate>

</DescriptionSetProfile>
```

6.3. Example 3: Constraining the value

The following DSP constrains the value to be a literal without a language.

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

<DescriptionSetProfile xmlns:foaf="http://xmlns.com/foaf/0.1/">

  <DescriptionTemplate ID="person" minOccurs="1" maxOccurs="1" standalone="yes">

    <ResourceConstraint>
      <MemberOf>
        <Option value="foaf:Person"/>
      </MemberOf>
    </ResourceConstraint>

    <StatementTemplate minOccurs="1" maxOccurs="1" type="literal">
      <PropertyConstraint>
        <InSet>
          <Option value="foaf:name"/>
        </InSet>
      </PropertyConstraint>

      <LiteralValueConstraint>
        <LanguageConstraint occurrence="disallowed"/>
      </LiteralValueConstraint>

    </StatementTemplate>

  </DescriptionTemplate>

</DescriptionSetProfile>
```

6.4. Example 4: Two resources

The following DSP allows for two kinds of resources: a single document, and multiple authors. The Person resources may only occur as values of the `dcterms:creator` property, not stand-alone. The value may only be described in a separate description with a mandatory `foaf:name` property.

```

```

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

<DescriptionSetProfile xmlns:foaf="http://xmlns.com/foaf/0.1/"
                       xmlns:dcterms="http://purl.org/dc/terms/">

  <DescriptionTemplate ID="document" minOccurs="1" maxOccurs="1" standalone="yes">

    <ResourceConstraint>
      <MemberOf>
        <Option value="dcterms:Text"/>
      </MemberOf>
    </ResourceConstraint>

    <StatementTemplate minOccurs="1" type="nonliteral">
      <PropertyConstraint>
        <InSet>
          <Option value="dcterms:creator"/>
        </InSet>
      </PropertyConstraint>

      <NonLiteralConstraint description="person">
        <ValueURIConstraint occurrence="disallowed"/>
        <VESConstraint occurrence="disallowed"/>
        <ValueStringConstraint maxOccur="0"/>
      </NonLiteralConstraint>

    </StatementTemplate>

  </DescriptionTemplate>

  <DescriptionTemplate ID="person" standalone="no">

    <ResourceConstraint>
      <MemberOf>
        <Option value="foaf:Person"/>
      </MemberOf>
    </ResourceConstraint>

    <StatementTemplate minOccurs="1" maxOccurs="1" type="literal">
      <PropertyConstraint>
        <InSet>
          <Option value="foaf:name"/>
        </InSet>
      </PropertyConstraint>

      <LiteralValueConstraint>
        <LanguageConstraint occurrence="disallowed"/>
      </LiteralValueConstraint>

    </StatementTemplate>

  </DescriptionTemplate>

</DescriptionSetProfile>
```

7. Description Templates

A description Template has the following attributes.

7.1. Identifier

Summary

A string that can be used in a Value Constraint to reference a description template that applies to the value resource.

Allowed values

A valid XML ID string.

Default

N/A

7.2. Standalone

Summary

Whether descriptions matching this template are allowed to occur standalone, i.e. without being the value of a property.

Allowed values

"yes" / "no" / "both"

Default

"both"

Conditions

If standalone is "yes", a matching description may not be a description of value occurring elsewhere in the DSP.
If standalone is "no", a matching description *must* be a description of value occurring elsewhere in the DSP.
If standalone is "both", both are allowed.
If this description template is referred to in a Value Constraint, standalone cannot be "yes".

7.3. Minimum occurrence constraint

Summary

The minimum number of times this kind of description must appear in the Description Set.

Allowed values

non-zero integer

Default

0

Conditions

must be equal or less than the Maximum occurrence

7.4. Maximum occurrence constraint

Summary

The maximum number of times this kind of description is allowed to appear in the Description Set.

Allowed values

non-zero integer or "infinity"

Default:'

"infinity"

Conditions

must be equal or less than the Minimum occurrence

7.5. Resource Class Membership Constraint

Summary

Classes that the resource may be an instance of

Allowed values

a list of class URIs

Default

no constraint

Conditions

if given, the resource must be an instance of one of the given classes.

8. Statement templates

A statement template has the following possible constraints.

8.1. Minimum occurrence constraint

Summary

The minimum number of times this kind of statement must appear in the enclosing Description.

Allowed values

non-zero integer

Default

0

Conditions

must be equal or less than the Maximum occurrence

8.2. Maximum occurrence constraint

Summary

The maximum number of times this kind of statement is allowed to appear in the enclosing Description.

Allowed values

non-zero integer or "infinity"

Default:'

"infinity"

Conditions

must be equal or less than the Minimum occurrence

8.3. Type constraint

Summary

The type of value surrogate (literal/non-literal) that is allowed in this Statement.

Allowed values

"literal" / "nonliteral"

Default

both allowed

Conditions

If no value is given, no further constraining on the value surrogate can be made.

Note: that the type constraint should follow any range given for the used properties.

8.4. Property constraints

There are two ways of constraining the property in a statement:

- By giving an explicit list of allowed properties
- By requiring the property to be a sub-property of a given property.

Exactly one of the above methods must be used in a single statement template.

8.4.1. Property list constraint

Summary

A set of properties that are allowed in this statement template.

Allowed values

a list of property URIs

Default

N/A

Conditions

cannot occur together with a sub-property constraint

8.4.2. Sub-property constraint

Summary

Only sub-properties of the given property are allowed in this statement template. Note that the given property is included in this list (all properties are sub-properties of themselves).

Allowed values

a property URI

Default

N/A

Conditions

cannot occur together with a property list constraint

8.5. Literal value constraints

Constrains a literal value surrogate in a statement. Only allowed in the case that the type constraint has the value "literal".

8.5.1. Literal list constraint

Summary

Literals that are allowed as values.

Allowed values

a list of literals, i.e. (string, language tag) or (string, syntax encoding scheme URI) pairs.

Default

no constraint

Conditions

if given, no other literal constraint may be given

8.5.2. Literal language constraint

2007-08-25 Usage Board meeting, Singapore

Summary

Whether languages are allowed for the literal

Allowed values

"mandatory" / "optional" / "disallowed"

Default

"optional"

Conditions

if "mandatory", Syntax encoding schemes are automatically disallowed.

8.5.3. Literal language list constraint

Summary

Languages allowed for the literal

Allowed values

a list consisting of language tags

Default

no constraint

8.5.4. Syntax Encoding Scheme constraint

Summary

Whether Syntax Encoding Scheme are allowed for the literal

Allowed values

"mandatory" / "optional" / "disallowed"

Default

"optional"

Conditions

if "mandatory", language tags are automatically disallowed.

8.5.5. Syntax Encoding Scheme list constraint

Summary

Syntax encoding schemes allowed for the literal

Allowed values

a list consisting of syntax encoding scheme URIs

Default

no constraint

8.6. Non-literal value constraints

Constrains the value surrogate in a statement. Only allowed in the case that the type constraint has the value "nonliteral".

8.6.1. Description template reference

Summary

A reference to a description template that may be used to describe the value

Allowed values

an identifier defined in a Description Template

Default

Related description not allowed

Conditions

if given, any related description of the value within the record must match the referenced Description Template. If the referenced Description Template contains mandatory Statement templates, such a description of the value must exist.

8.6.2. Class membership constraint

Summary

Classes that the value may be an instance of

Allowed values

a list of class URIs

Default

no constraint

Conditions

if given, the value must be an instance of one of the given classes.

Note: this is not a syntactic constraint, and as such might not be evaluated by all processors. If a type statement is desired, an explicit Statement template in a Description Template for the value resource should be created.

8.6.3. Value URI constraint

8.6.3.1. Value URI occurrence constraint

Summary

Whether a value URI must be given

Allowed values

"disallowed" / "optional" / "mandatory"

Default

"optional"

Conditions

8.6.3.2. Value URI list constraint

Summary

URIs that are allowed as value URIs.

Allowed values

a list of URIs

Default

no constraint

Conditions

If a value URI is given, it must be taken from this list. Cannot be specified if value occurrence is "disallowed"

8.6.4. Vocabulary encoding scheme constraint

8.6.4.1. Vocabulary encoding scheme occurrence constraint

Summary

Whether a vocabulary encoding scheme must be given

Allowed values

"disallowed" / "optional" / "mandatory"

Default

"optional"

Conditions

8.6.4.2. Vocabulary encoding scheme list constraint

Summary

URIs that are allowed as Vocabulary Encoding schemes.

Allowed values

a list of URIs

Default

no constraint

Conditions

If a vocabulary encoding scheme is given, it must be taken from this list. Cannot be specified if vocabulary encoding scheme occurrence is "disallowed"

8.6.5. Value String Constraints

If at least one value string constraint is given, any value string must match at least one of the constraints. If no value string constraint is given, any value string is allowed.

For each value string constraint, the following may be specified.

8.6.5.1. Minimum occurrence constraint

Summary

The minimum number of times this kind of value string must appear in the enclosing Statement.

Allowed values

non-zero integer

Default

0

Conditions

must be equal or less than the Maximum occurrence

8.6.5.2. Maximum occurrence constraint

Summary

The maximum number of times this kind of value string is allowed to appear in the enclosing Statement.

Allowed values

non-zero integer or "infinity"

Default:'

"infinity"

Conditions

must be equal or less than the Minimum occurrence

8.6.5.3. Other constraints

All Literal constraints can be used for value strings as well.

9. XML structure

- First version of XML schema

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

<DescriptionSetTemplate>

  <DescriptionTemplate standalone="" ID="" minOccur="" maxOccur="">

    <ResourceClass></ResourceClass>
    <ResourceClass></ResourceClass>

    <StatementTemplate ID="" minOccur="" maxOccur="" type="">

      <Property></Property>
      <Property></Property>

      <SubPropertyOf></SubPropertyOf>

      <NonliteralConstraint descriptionTemplateID="">

        <ValueClass></ValueClass>
        <ValueClass></ValueClass>

        <ValueURIOccurrence></ValueURIOccurrence>

        <ValueURI></ValueURI>
        <ValueURI></ValueURI>

        <VocabularyEncodingSchemeOccurrence></VocabularyEncodingSchemeOccurrence>

        <VocabularyEncodingScheme></VocabularyEncodingScheme>
        <VocabularyEncodingScheme></VocabularyEncodingScheme>

        <ValueStringConstraint  minOccur="" maxOccur="">
          <LiteralOption lang="" SES=""></LiteralOption>
          <LiteralOption lang="" SES=""></LiteralOption>

          <LanguageOccurrence></LanguageOccurrence>

          <Language></Language>
          <Language></Language>

          <SyntaxEncodingSchemeOccurrence></SyntaxEncodingSchemeOccurrence>

          <SyntaxEncodingScheme></SyntaxEncodingScheme>
          <SyntaxEncodingScheme></SyntaxEncodingScheme>

        </ValueStringConstraint>
```

```
      </NonLiteralConstraint>

      <LiteralConstraint>
          <LiteralOption lang="" SES=""></LiteralOption>
          <LiteralOption lang="" SES=""></LiteralOption>

          <LanguageOccurrence></LanguageOccurrence>

          <Language></Language>
          <Language></Language>

          <SyntaxEncodingSchemeOccurrence></SyntaxEncodingSchemeOccurrence>

          <SyntaxEncodingScheme></SyntaxEncodingScheme>
          <SyntaxEncodingScheme></SyntaxEncodingScheme>

      </LiteralConstraint>

    </StatementTemplate>

  </DescriptionTemplate>

</DescriptonSetProfile>
```

10. RDF variant

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:dsp="http://purl.org/dc/dsp/">

  <dsp:DescriptionTemplate rdf:about="#d1">
    <dsp:standalone rdf:datatype="xsd:boolean">true</dsp:standalone>
    <dsp:minOccur rdf:datatype="xsd:nonNegativeInteger">0</dsp:minOccur>
    <dsp:maxOccur rdf:datatype="xsd:nonNegativeInteger">0</dsp:maxOccur>

    <dsp:resourceClass rdf:resource=""/>
    <dsp:resourceClass rdf:resource=""/>

    <dsp:statementTemplate>
      <dsp:LiteralStatementTemplate>
        <dsp:minOccur rdf:datatype="xsd:nonNegativeInteger">0</dsp:minOccur>
        <dsp:maxOccur rdf:datatype="xsd:nonNegativeInteger">0</dsp:maxOccur>

        <dsp:property rdf:resource=""/>
        <dsp:property rdf:resource=""/>

        <dsp:subPropertyOf rdf:resource=""/>

        <dsp:literalConstraint>
          <dsp:LiteralConstraint>
            <dsp:literal xml:lang="" rdf:datatype=""></dsp:literal>
            <dsp:literal xml:lang="" rdf:datatype=""></dsp:literal>

            <dsp:languageOccurrence rdf:datatype="dsp:occurrence"></dsp:languageOccurrence>

            <dsp:language rdf:datatype="xsd:language"></dsp:language>
            <dsp:language rdf:datatype="xsd:language"></dsp:language>

            <dsp:syntaxEncodingSchemeOccurrence rdf:datatype="dsp:occurrence"></dsp:syntaxEncodingSchemeOccurrence>

            <dsp:syntaxEncodingScheme rdf:resource=""/>
            <dsp:syntaxEncodingScheme rdf:resource=""/>

          </dsp:LiteralConstraint>
        </dsp:literalConstraint>
      </dsp:LiteralStatementTemplate>
    </dsp:statementTemplate>

    <dsp:statementTemplate>
      <dsp:NonLiteralStatementTemplate>
         <dsp:nonLiteralConstraint>
          <dsp:NonLiteralConstraint>

            <dsp:descriptionTemplate rdf:resource=""/>

            <dsp:valueClass rdf:resource=""/>
            <dsp:valueClass rdf:resource=""/>

            <dsp:valueURIOccurrence rdf:datatype="dsp:occurrence"></dsp:valueURIOccurrence>

            <dsp:valueURI rdf:datatype="xsd:URI"></dsp:valueURI>
            <dsp:valueURI rdf:datatype="xsd:URI"></dsp:valueURI>
```

```
            <dsp:vocabularyEncodingSchemeOccurrence rdf:datatype="dsp:occurrence"></dsp:
vocabularyEncodingSchemeOccurrence>

            <dsp:vocabularyEncodingScheme rdf:resource=""/>
            <dsp:vocabularyEncodingScheme rdf:resource=""/>

            <dsp:valueStringConstraint>
              <dsp:ValueStringConstraint>

                <dsp:minOccur rdf:datatype="xsd:nonNegativeInteger">0</dsp:minOccur>
                <dsp:maxOccur rdf:datatype="xsd:nonNegativeInteger">0</dsp:maxOccur>

                <dsp:literal xml:lang="" rdf:datatype=""></dsp:literal>
                <dsp:literal xml:lang="" rdf:datatype=""></dsp:literal>

                <dsp:languageOccurrence rdf:datatype="dsp:occurrence"></dsp:languageOccurrence>

                <dsp:language rdf:datatype="xsd:language"></dsp:language>
                <dsp:language rdf:datatype="xsd:language"></dsp:language>

                <dsp:syntaxEncodingSchemeOccurrence rdf:datatype="dsp:occurrence"></dsp:syntaxEncodingSchemeOccurrence>

                <dsp:syntaxEncodingScheme rdf:resource=""/>
                <dsp:syntaxEncodingScheme rdf:resource=""/>

              </dsp:ValueStringConstraint>
            </dsp:valueStringConstraint>
          </dsp:NonLiteralConstraint>
        </dsp:nonLiteralConstraint>
      </dsp:NonLiteralStatementTemplate>
    </dsp:statementTemplate>
  </dsp:DescriptionTemplate>
</rdf:RDF>
```

11. Examples

Simple DC:

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

<DescriptionSetTemplate xmlns:dcterms="http://purl.org/dc/terms/">

  <DescriptionTemplate>

    <StatementTemplate>
      <PropertyConstraint>
        <InSet>
          <Option value="dcterms:title"/>
        </InSet>
      </PropertyConstraint>
    </StatementTemplate>

    <StatementTemplate>
      <PropertyConstraint>
        <InSet>
          <Option value="dcterms:description"/>
        </InSet>
      </PropertyConstraint>
    </StatementTemplate>

   <!-- etc -->

  </DescriptionTemplate>

</DescriptonSetProfile>
```

12. Simple FOAF

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

<DescriptionSetTemplate xmlns:foaf="http://xmlns.com/foaf/0.1/">

  <DescriptionTemplate ID="person" minOccurs="1" maxOccurs="1">

    <ResourceConstraint>
      <MemberOf>
        <Option value="foaf:Person"/>
      </MemberOf>
```

```
    </ResourceConstraint>

    <StatementTemplate minOccurs="1" maxOccurs="1" type="literal">
      <PropertyConstraint>
        <InSet>
          <Option value="foaf:name"/>
        </InSet>
      </PropertyConstraint>
    </StatementTemplate>

    <StatementTemplate type="nonliteral">
      <PropertyConstraint>
        <InSet>
          <Option value="foaf:knows"/>
        </InSet>
      </PropertyConstraint>
      <NonLiteralConstraint decription="person">
        <MemberOf>
          <Option value="foaf:Person"/>
        </MemberOf>
      </NonLiteralConstraint>
    </StatementTemplate>

   <!-- etc -->

  </DescriptionTemplate>

</DescriptonSetProfile>
```

Introduction

This document describes a DC Application Profile for describing an eprint. The application profile is based on the Eprints Model, which is in turn based on FRBR. The model comprises 5 entities - ScholarlyWork, Expression, Manifestation, Copy and Agent. This application profile provides a way of describing these entities as part of a *description set* (a set of related DC *descriptions*).

The notion of a *description set*
is part of the DCMI Abstract Model. Readers that are not familiar with the DCMI Abstract Model should read A note about the DCMI Abstract Model before proceeding.

All the examples in this document are formatted according to the DC-Text notation. Readers who are not familiar with DC-Text should read A note about DC-Text before proceeding.

In the context of this work an eprint is defined to be a *scientific or scholarly research text* (as defined by the Budapest Open Access Initiative), for example a peer-reviewed journal article, a preprint, a working paper, a thesis, a book chapter, a report, etc.

Eprints Application Profile

Each *description set*
that complies with the Eprints Application Profile is made up of a set of related *descriptions* about the entities listed above. Therefore, a typical *description set* (using the DC-Text notation) has the following structure:

```
Description Set (
    Description (
        # description of the eprint as a ScholarlyWork
        ...
    )
    Description (
        # description of an Expression of the eprint
        ...
    )
    Description (
        # description of a Manifestation of an Expression of the
        ...
    )
    Description (
        # description of a Copy of a Manifestation of an Expressi
        ...
    )
    Description (
```

```
        # description of an author, funder, supervisor of the epr
        ...
    )
    Description (
        # description of an editor of an Expression of the eprint
        ...
    )
    Description (
        # description of the publisher of a Manifestation of an E
        ...
    )
    ...
)
```

Each *description set* describes only one eprint (i.e. one ScholarlyWork entity). However, multiple *descriptions*
may be used to describe multiple Expression, Manifestation and Agent entities as necessary.

The metadata terms that may be used to describe each entity are described below. Note that all *properties* may be repeated if necessary. Unless otherwise noted, multiple *value strings* should be provided using separate *statements*. Where appropriate, each *value string* may have an associated *language tag*.

A minimal *description set* that conforms to this application profile must include either:

- a single ScholarlyWork *description* with at least one dc:title *statement* and one dc:type *statement* indicating that this is an entity of type http://purl.org/eprint/entityType/ScholarlyWork, or
- a single ScholarlyWork *description* with one one dc:type *statement* indicating that this is an entity of type http://purl.org/eprint/entityType/ScholarlyWork and one eprints:isExpressedAs *statement* linking to a single Expression *description* with at least one dc:title *statement* and one dc:type *statement* indicating that this is an entity of type http://purl.org/eprint/entityType/Expression.

All other aspects of the application profile are optional.

Description of the eprint as a ScholarlyWork

Entity type

Property	http://purl.org/dc/elements/1.1/type
Min occurrence	1
Max occurrence	1
Literal?	No
Definition	The type nature or genre of the content of the resource.

Eprint-specific recommendation	Each of the entity *descriptions* in the *description sets* conforming with this application profile will need to be explicitly typed. This is done using a dc:type *statement* with one of the following *value URIs* taken from the Eprints EntityType Vocabulary Encoding Scheme corresponding to the entity being described.
Value (Non-Literal)	**Value URI Constraint**: **Occurrence**: mandatory **Choose from**: http://purl.org/eprint/entityType/ScholarlyWork/ **Vocabulary Encoding Scheme Constraint** **Occurrence**: mandatory **Choose from**: http://purl.org/eprint/entityType/ **Value String Constraint:** Max occurrence: 0 *Syntax Encoding Syntax Constraint*: **Occurrence**: disallowed *Language Constraint*: **Occurrence**: disallowed

For example:

```
Statement (
    Property URI ( dc:type )
    Vocabulary Encoding Scheme URI ( eprint:EntityType )
    Value URI ( <http://purl.org/eprint/entityType/ScholarlyWork>
)
```

Title

Property	http://purl.org/dc/elements/1.1/title
Min occurrence	1
Literal?	Yes
Definition	A name given to the resource.
Eprint-specific recommendation	The title of the eprint. Preserve the original wording, order and spelling of the eprint title. In general, only capitalize proper nouns, though there may be exceptions to this rule particularly regarding Internet-related terms.

	Punctuation need not reflect the usage of the original. Subtitles should be separated from the title by 'space-colon-space'.

For example:

```
Statement (
    Property URI ( dc:title )
    Value String ( "Initial sequencing and analysis of the human
)
Statement (
    Property URI ( dc:title )
    Value String ( "New nationalism and the old history : perspec
)
```

Subject

Property	http://purl.org/dc/elements/1.1/subject
Literal?	No
Definition	The topic of the content of the resource.
Eprint-specific recommendation	The topic of the eprint. In general, choose the most significant and unique words for keywords, avoiding those too general to describe a particular eprint. For free-text keywords use multiple *statements*, one for each term. There are no requirements regarding the capitalization of keywords though internal (within archive) consistency is recommended. Where terms are taken from a standard classification scheme encode each term in a separate *statement*. Use a *vocabulary encoding scheme URI* to indicate the classification scheme in use. Encode the complete subject descriptor according to the relevant scheme. Use the capitalisation and punctuation used in the original scheme. Where subject terms are taken from LCSH, the subfields of the subject heading should be separated by double dash (--) and spaces should be omitted. If the subject of the eprint is a person or an organization, provide their name as a *value string* according to A note about the form of personal and organisational names used in value strings.
Value (Non-Literal)	**Value URI Constraint**: **Occurrence** optional **Vocabulary Encoding Scheme Constraint**

Occurrence:	optional
Choose from:	http://purl.org/dc/terms/LCSH http://authorities.loc.gov/

Value String Constraint:	
Max occurrence	1
Syntax Encoding Syntax Constraint:	
Occurrence:	disallowed
Language Constraint:	
Occurrence	optional

For example:

```
Statement (
    Property URI ( dc:subject )
    Value String ( "polar oceanography" )
)
Statement (
    Property URI ( dc:subject )
    Value String ( "boundary current" )
)
```

Using LCSH subjects:

```
Statement (
    Property URI ( dc:subject )
    Vocabulary Encoding Scheme URI ( dcterms:LCSH )
    Value String ( "World War, 1939-1945--Germany" )
)
Statement (
    Property URI ( dc:subject )
    Vocabulary Encoding Scheme URI ( dcterms:LCSH )
    Value String ( "Germany--History--1933-1945" )
)
```

Name as subject:

```
Statement (
    Property URI ( dc:subject )
    Value String ( "Hitler, Adolf, 1889-1945" )
)
```

Abstract

2007-08-25 Usage Board meeting, Singapore

```
        Vocabulary Encoding Scheme URI ( dcterms:RFC3066 )
        Value String ( "fr" )
    )
```

Type

Property	http://purl.org/dc/elements/1.1/type
Literal?	No
Definition	The nature or genre of the content of the resource.
Eprint-specific recommendation	The type of the described expression of the eprint. Recommended best practice is to provide a value URI for a class from the Eprints Type Vocabulary Encoding Scheme.
Value (Non-Literal)	**Value URI Constraint**: **Occurrence**: mandatory **Choose from**: http://purl.org/eprint/type/ScholarlyText http://purl.org/eprint/type/Book http://purl.org/eprint/type/BookItem http://purl.org/eprint/type/BookReview http://purl.org/eprint/type/ConferenceItem http://purl.org/eprint/type/ConferencePaper http://purl.org/eprint/type/ConferencePoster http://purl.org/eprint/type/JournalItem http://purl.org/eprint/type/JournalArticle http://purl.org/eprint/type/NewsItem http://purl.org/eprint/type/Patent http://purl.org/eprint/type/Report http://purl.org/eprint/type/SubmittedJournalArticle http://purl.org/eprint/type/Thesis http://purl.org/eprint/type/WorkingPaper **Vocabulary Encoding Scheme Constraint** **Occurrence**: mandatory **Choose from**: http://purl.org/eprint/type/ **Value String Constraint:** Max occurrence: 0 *Syntax Encoding Syntax Constraint*: **Occurrence**: disallowed *Language Constraint*:

Occurrence	disallowed

For example:

```
Statement (
    Property URI ( dc:type )
    Vocabulary Encoding Scheme URI ( eprint:Type )
    Value URI ( <http://purl.org/eprint/type/JournalArticle> )
)
```

Copyright Holder

Property	http://purl.org/eprint/terms/copyrightHolder
Literal?	No
Definition	A person or organization owning copyright in the resource.
Eprint-specific recommendation	A person or organization owning copyright in the eprint. Use this *property* to provide the copyright holder's name and/or the URI of the copyright holder and/or to link to a *related description* (with the *description set*) about the copyright holder. Where a name is provided, see A note about the form of personal and organisational names used in value strings.
Value (Non-Literal)	**Description**: agent **Vocabulary Encoding Scheme Constraint** **Occurrence**: optional **Choose from**: http://authorities.loc.gov/ **Value String Constraint:** Max occurrence: 1 *Syntax Encoding Syntax Constraint*: **Occurrence**: disallowed *Language Constraint*: **Occurrence**: optional

For example:

```
Statement (
    Property URI ( eprint:copyrightHolder )
    Value String ( "University of Leeds" )
```

2007-08-25 Usage Board meeting, Singapore

From KMR Wiki

Main: Draft: Wiki format for Description Set Profile

This is a draft document on how a Description Set Profile **[1]** is expressed in a Wiki-format. A Description Set Profile (abbreviated DSP in the rest of this document) is not directly intended for human consumption, but with a DSP expressed in a wiki-format it is possible to mix normal wiki-syntax with specific DSP expressions in order to document it, from that source it is possible to create:

- The Description Set Profile rendered as an HTML-page for human consumption
- A formal serialization in XML for the same Description Set Profile

The DSP wiki-syntax and how to use is described in this document and is focused on an implementation done for the moinmoin wiki engine **[2]** with the help of an extension to that wiki-engine. The DSP wiki-syntax itself is independent of wiki engine and is possible to use on another wiki-engine. This of course dependent on if and how the wiki engine of your choice supports extension. The extension in the moinmoin wiki-engine to render a HTML-page for the DSP, described here, has been done by adding a parser extension to the wiki. To call that parser, named DSP, inside a wiki-page the following syntax is used:

```
{{{#!DSP
 /* Here the specific Description Set Profile wiki-syntax to be parsed */
}}}
```

Anything above and below is parsed according to the normal moinmoin wiki syntax. For more information about parsers in moinmoin-wiki visit **http://moinmoin.wikiwikiweb.de/HelpOnParsers**

To create the DSP XML a so called action extension for the moinmoin wiki has been implemented, this will be described in the end of this document.

Design and description of the wiki syntax

As stated in the description of Description Set Profile **[1]**, it uses the notion of *constraint* and *templates* to describe the structure of a Description Set. There are two levels of templates in the DSP:

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

These templates furthermore consist of constraints that are used to limit the structure. The relations between the templates and the constraints are depicted in the figure 1 below

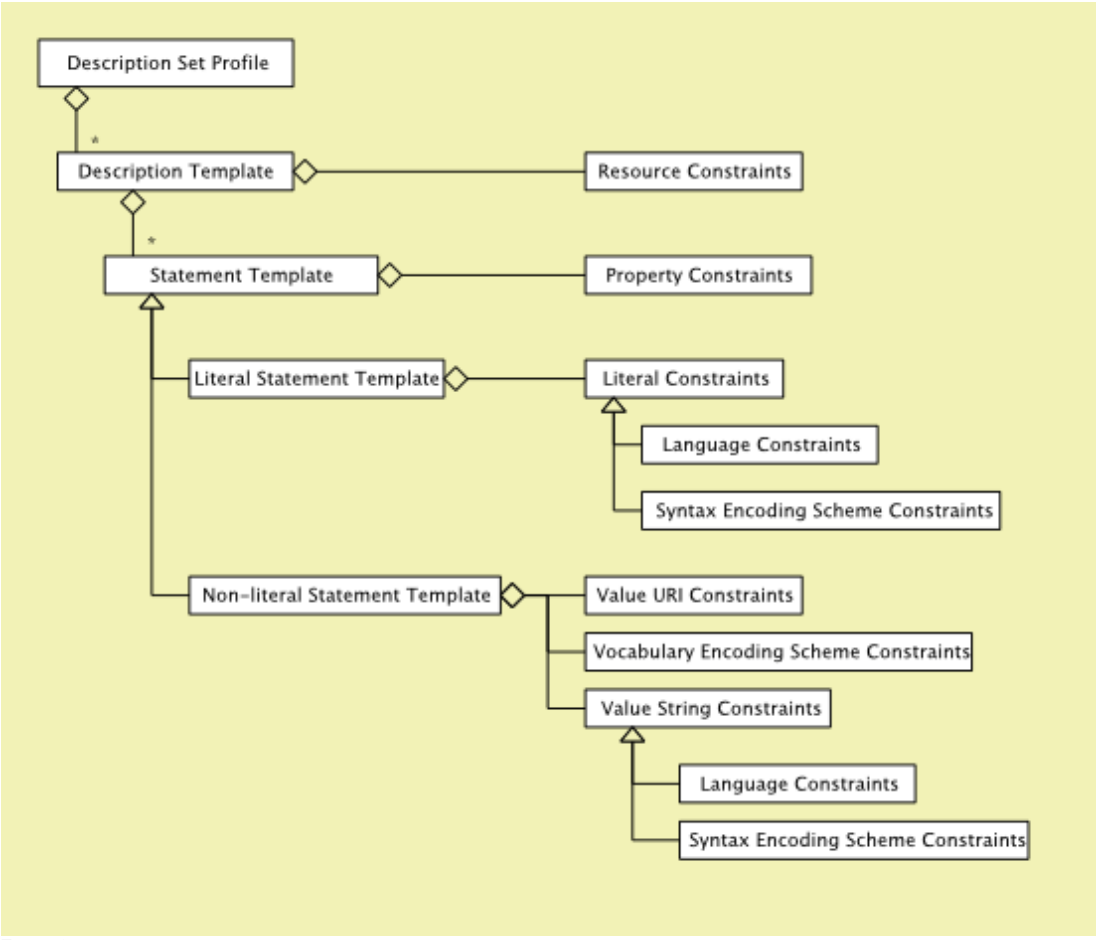


Figure 1

The major design choice was to break up the different parts, ie the templates and the constraint, that make up a Description Set Profile and make each part of them as easy to

express as possible. The hierarchical structure of the DSP is reused in the syntax in such a way that when a template or constraint has been declared it is a part of the first declared template above that it can be a part of according to the hierarchy. Ie if first a Description Template has been declared followed by a Statement Template, this Statement Template is seen as a part of that Description Template declared above it.

In the syntax 4 parts are possible to express on their own:

- **Description template**, directly corresponds to the description of it above.
- **Statement Template**, that also corresponds to the description above. However, a Statement can be either a *Literal Statement Template* or *Non-Literal Statement Template*. Of which type it is can be expressed inside.
- **Non-Literal Value Constraint**, this corresponds to the value constraints for a *Non-Literal Statement Template*
- **Literal Value Constraint** this corresponds to the value constraints for a *Literal Statement Template*

Before describing the syntax in more detail, below is an empty template to serve as an example of how the structure looks like:

```
{{{#!DSP

Any (moinmoin)wiki-format text, will be placed over the Description Template declared below when rendered
DT=(ID="" RC=[] min="" max="")

Any (moinmoin)wiki-format text, that is placed above the Statement Template below when rendered
ST=(ID="" min="" max="" type="" PC=(""|{,}))

NLC=(description="" []
        VURIConstraint=( occurrence="mandatory|optional|disallowed" {,})
        VESConstraint=( occurrence="mandatory|optional|disallowed" {,})
        VStringConstraint=(min="" max="" {[value="" lang="" SES=""], [...]}
                          LangC=(occurrence="mandatory|optional|disallowed" {,})
                          SESConstraint=(occurrence="mandatory|optional|disallowed" {,})
                         )
       )

ST=(ID="" min="" max="" type="" PC=(""|{,}))

LC=({[value="" lang="" SES=""], [...]}
       LangC=(occurrence="mandatory|optional|disallowed" {,})
       SESConstraint=(occurrence="mandatory|optional|disallowed" {,})
      )
)

ST=(...)


DT=(...)

ST=(...)

ST=(...)
}}}
```

In between each expressed DT, ST, NLC and LC the normal moinmoin wiki-syntax can be used.

Some small remarks on the usage of symbols in general in the format, when something in the format can have one and only on value it is placed between citation marks, as an example, see identifier for Description Template below. If the a set of choices is to be expressed the curly brackets are used in most places.

Expressing the different parts of a DSP

As stated above and seen in the template above the 4 basic parts are expressed with:

- **DT** for a Description Template
- **ST** for a Statement Template
- **NLC** for a NonLiteral Value Constraint
- **LC** for a Literal Value Constraint.

Each one of these expression has to start on a new line. As can also be seen from the template above it is possible to express normal wiki-syntax in between. This text does also need to start on a new line in order to not be mix up with the DSP wiki-syntax, this text is surrounding the parts of a DSP in the resulting HTML, but will be ignored when creating the XML for the DSP. This means that the HTML that is created for example an ST can be embedded into a table or some similar construct and the main reason for using a wiki, since you both construct the DSP and its documentation. If the keywords DT, ST, NLC or LC needs to start on an new line in the normal wikisyntax, add a backslash ("\") before and the backslash will be ignored in the resulting HTML.

Expressing a Description Template

Start a new line with "DT" followed by "=" and a starting left parenthesis and in order to close the declaration a right parenthesis is needed. The expression inside the parenthesis is allowed to span over more than one line. Expressed inside the parenthesis are the attributes:

- **Identifier**, use the keyword "ID" followed by a "=" and a valid XML ID String inside citation marks, example expression: ID="Person"
- **Resource Class Membership Constraint**, is expressed in a list with the keyword RC followed by "=" and a list with the URIs inside angle brackets separated by a comma and a blankspace. Example expression: RC=[**http://www.anyuri.net/someResourceConstraint**, **http://www.anyuri.net/someOtherResourceConstraint**]
- **Minimum occurrence constraint**, use the keyword "min" followed by a "=" with a non negative integer inside citation mark as the value, example expression: min="1"

- **Maximum occurrence constraint**, use the keyword "max" followed by a "=" with a non negative integer inside citation mark as the value, example expression: max="1"

For a more detailed description about these attributes see the section about Description Template in the description of DSP **[DT]**. Here conditions for the attributes and possible and default values are described in more detail.

The attributes of a Description Template can be declared inside the parenthesis in any order with blankspace as separator. If none of them are declared the parenthesis should be left empty. The Statment Templates that belong to a Description Template are the ones that are declared under until a new declaration of a Description Template occurrs.

Example of a declaration of a Description Template:

```
  DT=(ID="Person" RC=[http://www.anyuri.net/someResourceConstraint, http://www.anyuri.net/someOtherResourceConstraint]
min="1" max="1")
```

Expressing a Statement Template

A Statement Template is expressed in a similar way to the Description Template, starting on a new line with "ST=" followed by a left parenthesis and then closed by a right parenthesis. The expression inside the parenthesis is allowed to span over more than one line and the attributes inside are separated with a blankspace and is allowed to be in any order. Expressed inside the parenthesis are the attributes:

- **Property list constraint** or **Sub-property constraint**, to declare this start with the keyword "PC" followed by "=" and one and only one of the following:
  - on URI within surrounding citation marks to declare a Sub Property Constraint, example expression: PC="**http://purl.org/dc/elements/1.1/creator**"
  - a set of URI:s, by using curly braces with comma and a blankspace as a separator to declare a Property List Constraint. Example expression: PC={**http://purl.org/dc/elements/1.1/creator**, **http://xmlns.com/foaf/0.1/name**}
- **Minimum occurrence constraint**, use the keyword "min" followed by a "=" with a non negative integer inside citation mark as the value, example expression: min="1"
- **Maximum occurrence constraint**, use the keyword "max" followed by a "=" with a non negative integer inside citation mark as the value, example expression: max="1"
- **Type Constraint**, is declared with the keyword "type" followed by a "=" and then either "*Literal*" or "*NonLiteral*" inside citation marks. The value of this attribute decides if a Statement Template is of type Literal Statement Template or Non-literal Statement Template. If the attribute is not declared it will default to be both and no further value constraints on can be made according to the description of DSP **[Statement Template type]**, Example: type="literal"

For a more detailed description about these attributes see the section about Statement Template in the description of the DSP **[ST]**, where conditions for the attributes and possible and default values are described in more detail. Example of the parts described above:

```
 ST=(min="1" max="10" PC="http://purl.org/dc/elements/1.1/creator" type="literal")
```

A Statement Template can be of either the type Literal Statement Template and have a Literal Value Constraint or of the type Non-literal Statement Template and have a Non-literal Value Constraint. These constraints correspond to the *Literal value surrogates* and *Non-literal value surrogates* respectively in the **Dublin Core Abstract Model**. To express constraints on a Statement Template they will be declared under the Statement Template that they belong to and they also need to start on a new line.

Expressing a literal value constraint

When an Statement Template has got the type constraint set to Literal it is possible to further constrain the possible values, ie a Literal Value Constraint. To express this constraint the keyword "LC" followed by a "=" and a left parenthesis is used, at the end a right parenthesis is used to close the expression. The declaration of a Literal value constraint have to start on a new line. Inside the parenthesis the following parts can be expressed:

- **Constraints on Syntax Encoding Scheme**, a set of such constraint are declared with the keyword "SESConstraint" followed by a "=" and a left parenthesis, a right parenthesis close the expression. Inside the parenthesis the following can be declared:
  - **Syntax Encoding Scheme constraint** that decides the occurrence of a Syntax Encoding Scheme constraint, starts with the keyword "occurrence" that have the possible values of *mandatory*, *optional* or *disallowed*. If this part is not declared it will by default be "optional", example syntax: occurrence="mandatory"
  - **Syntax Encoding Scheme list constraint**, that states the syntax encoding schemes allowed for the literal, declared as a set of options (in the form of a URI) inside curly brackets, separated by a comma and a blankspace. Example syntax: {**http://purl.org/dc/terms/URI**}
- **Literal list constraint**, are the literals that are allowed as values and are declared as a set inside curly braces, each entry inside the curly braces are declared inside angle brackets and are separated with a comma and a blank space. Inside each entry a value is declared with the keyword "value" followed by a "=" and the value as a string, after the value either a language or a Syntax Encoding Scheme can be declared with the keywords "lang" or "SES" used respectively followed by a "=" and the value. For the language a ISO language tag are the valid values, for Syntax Encoding Scheme a URI is a valid value. Example expression {[value="A simple String" lang="en"], [...]}
- **Constraints on the langugae**, are declared by using the keyword "LangC" followed by "=" and a pair of parenthesis that inside contain:
  - **Literal language constraint**, that decides the occurrence of a language constraint, starts with the keyword "occurence" that have the possible values of *mandatory*, *optional* or *disallowed*. If this part is not declared it will by default be "optional", example syntax: occurrence="mandatory"
  - **Literal language list constraint** are declared as a set of languages choices expressed inside curly braces with comma and a blankspace as a separator. If no constraint is to be given this part can be left out. Example expression: {en, sv, es}

For a more detailed description about the attributes for Literal Value Constraints see the section about this in the description of the DSP **[LC]**, where conditions on the attributes and possible and default values are described in more detail.

Example 1 of how to express an LC:
LC=({[value="SomeEncoding" lang="en" SES="**http://www.example.org/someSyntaxEncodingScheme**"]})

Example 2 of how to express an LC with languageConstraint:
LC=(LangC=(occurrence="optional" {en, sv, es}))

Expressing a non-literal Value Constraint

When an Statement Template has got the type constraint set to NonLiteral it is possible to further constrain the possible values, ie a Non-literal Value Constraint. To declare this

constraint start on a new line and use the keyword "NLC" followed by a "=" and a left parenthesis, at the end of the NLC a right parenthesis is used to close the expression. The expression is allowed to span over more than one line. Inside the parenthesis the following parts can be expressed:

- **Description template reference**, this is a reference to a description template that may be used to describe the value. Example expression: description="ref"
- **Class membership constraint**, classes that the value may be an instance of (in the form of a URI) and is declared inside angle-brackets with comma and a blankspace as a separator. Example: {**http://www.example.com/exampleURI1**, ...}
- **Value URI Constraint**, are expressed with the keyword "VURIConstraint" with a following "=" followed by a left parenthesis and it is closed with a right one. Inside two arguments are provided:
  - **Value URI list constraint**, URIs that are allowed as value URIs. Declared as a set and expressed inside curly braces with comma and a blankspace as a separator, Example expression: {**http://www.example.com/uri1**, **http://www.example.com/uri2**}
  - **Value URI occurrence constraint**, that have the possible values of *mandatory*, *optional* or *disallowed*. Example syntax: occurrence="mandatory".
  - Example of the ValueURIConstraint construct:

VURIConstraint=(occurrence="mandatory" {**http://www.example.com/uri1**, **http://www.example.com/uri2**})

- **Vocabulary encoding scheme constraint**, is used if a vocabulary encoding scheme must be given. This is declared almost the same way as ValueURIConstraint, but with the keyword "VESConstraint" used instead. Example
  VESConstraint=(occurrence="mandatory" {**http://www.example.com/uri3**, **http://www.example.com/uri4**})
- **Value String Constraint** are declared by using the keyword "VStringConstraint" and with its argument inside left and right parenthesis. Inside the following is declared:
  - **Minimum occurrence constraint**, use the keyword "min" followed by a "=" with a non zero Integer inside citation mark as the value, example expression: min="1"
  - **Maximum occurrence constraint**, use the keyword "max" followed by a "=" with a non zero Integer inside citation mark as the value, example expression: max="1"
  - Apart from the *maximum* and *minimum occurrence constraint* the rest of the declaration of a Value String Constraint are done the same way as a **Literal Constraint**

For a more detailed description about the attributes for Non-Literal Value Constraints see the section about this in the description of the DSP **[NLC]**, where conditions on the attributes and possible and default values are described in more detail.

Example 1 of an NLC

```
NLC=(description="ref" {http://www.example.com/exampleURI1})
```

Example 2 of an NLC with ValueURIConstraint

```
NLC=(description="ref" VURIConstraint=( occurrence="mandatory" {http://www.example.com/uri1, http://www.example.com/uri2}))
```

Example 3 of an NLC with VESConstraint

```
NLC=(description="ref" VESConstraint=( occurrence="mandatory" {http://www.example.com/uri3, http://www.example.com/uri4}))
```

Example 4 of an NLC with ValueStringConstraint

```
NLC=(description="ref" VStringConstraint=(min="1" max="1" {[value="someFormat:Format2" lang="en"
 SES="http://www.example.org/someSyntaxEncodingScheme"]}))
```

Examples

The DC-XML example from **http://dublincore.org/architecturewiki/DescriptionSetProfile** in wiki-format:

Example 1:

```
{{{#!DSP

DT=()

ST=(PC={http://purl.org/dc/terms/title})

ST=(PC={http://purl.org/dc/terms/description})

}}}
```

Example 2

```
{{{#!DSP

DT=( ID="Person" RC=[http://xmlns.com/foaf/0.1/Person] min=1 max=1 )

ST=( min=1 max=1 type="literal" PC={http://xmlns.com/foaf/0.1/name} )

ST=( type="literal" PC={http://xmlns.com/foaf/0.1/knows} )

NLC=({http://xmlns.com/foaf/0.1/Person})

}}}
```

2007-08-25 Usage Board meeting, Singapore

Usage example

The implementation of this syntax has been done as an extension to the the moinmoin wiki engine. Below follows an example and it will show how the syntax will be rendered in HTML and also how the resulting XML looks like.

Example of the syntax:

With the extension to the moinmoin wiki engine installed the syntax described in this document can be used on a page. It will be parsed as a Description Set Profile, below is an example:

```
{{{#!DSP
== Description template ==
DT=(min="1" max="1" standalone="yes")

=== Title ===
----
ST=(type="literal" PC={http://purl.org/dc/terms/title})
|| Definition || A name given to the resource. ||
LC=(LangC=(occurrence="optional") SESConstraint=(occurrence="disallowed") )

=== Creator ===
----
ST=(type="nonliteral" PC={http://purl.org/dc/terms/creator})
|| Definition || An entity primarily responsible for making the resource. ||
|| Comment || Examples of a Creator include a person, an organization, or a service.
Typically, the name of a Creator should be used to indicate the entity. ||
NLC=( VURIConstraint=( occurrence="disallowed") VESConstraint=( occurrence="disallowed" ) VStringConstraint=(max="1"
LangC=(occurrence="optional") SESConstraint=(occurrence="disallowed") ) )
}}}
```

The resulting HTML:

The example of the syntax above will display the following when the page is called:

Description template

Title

Property	**http://purl.org/dc/terms/title**
Literal?	Yes
Definition	A name given to the resource.
Value (Literal)	Syntax Encoding Scheme: Occurrence: disallowed Language Constraint: Occurrence: optional

Creator

Property	**http://purl.org/dc/terms/creator**
Literal?	No
Definition	An entity primarily responsible for making the resource.
Comment	Examples of a Creator include a person, an organization, or a service. Typically, the name of a Creator should be used to indicate the entity.
Value(Non-Literal)	Value URI Constraint: Occurrence: disallowed Vocabulary Encoding Scheme Constraint: Occurrence: disallowed Value String Constraint: -Syntax Encoding Syntax Constraint: Occurrence: "disallowed" -Language Constraint: Occurrence: "disallowed"

As can be seen here the Statement Template and the Literal and Non-Literal Value constraints are rendered in a table in HTML. That table can be integrated into another table declared in the normal wiki-syntax, as done above.

The resulting XML

2007-08-25 Usage Board meeting, Singapore

From the wiki-syntax an action-extension has been implemented in order to create XML from the DSP wiki syntax. The XML for a DSP can be retrieved by adding "?action=DSP2XML" at the end of the URL of the page that contains a DSP. Below is the XML for the example syntax given above:

```
<DescriptionSetTemplate>
  <DescriptionTemplate maxOccur="1" minOccur="1">
    <StatementTemplate type="literal">
      <Property>http://purl.org/dc/terms/title</Property>
      <LiteralConstraint>
        <SyntaxEncodingSchemeOccurrence>"disallowed"</SyntaxEncodingSchemeOccurrence>
        <LanguageOccurrence>"optional"</LanguageOccurrence>
      </LiteralConstraint>
    </StatementTemplate>
    <StatementTemplate type="nonliteral">
      <Property>http://purl.org/dc/terms/creator</Property>
      <NonliteralConstraint>
        <ValueURIOccurrence>disallowed</ValueURIOccurrence>
        <VocabularyEncodingSchemeOccurrence>disallowed</VocabularyEncodingSchemeOccurrence>
        <ValueStringConstraint maxOccur="1">
          <SyntaxEncodingSchemeOccurrence>"disallowed"</SyntaxEncodingSchemeOccurrence>
          <LanguageOccurrence>"optional"</LanguageOccurrence>
        </ValueStringConstraint>
      </NonliteralConstraint>
    </StatementTemplate>
  </DescriptionTemplate maxOccur="1" minOccur="1">
</DescriptionSetTemplate>
```

Simple Dublin Core - an application profile

Introduction

"The Dublin Core" was first defined in 1995 as a set of thirteen elements for resource description. The fifteen-element RFC 2413 "Dublin Core Metadata for Resource Discovery" of 1998 referred to "the simple Dublin Core scheme" [RFC2413]. The approval of a set of forty-one "qualifiers" in July 2000 led to a distinction between Simple Dublin Core(just the fifteen elements) and Qualified Dublin Core (the fifteen elements used with qualifiers).

With the approval of qualifiers, a draft "Using Dublin Core in XML" (July 2000) [DCMES-XML1], was renamed "An XML Encoding of Simple Dublin Core Metadata" (November 2000) [DCMES-XML2], then "Expressing Simple Dublin Core in RDF/XML" (September 2001) [DCMES-XML3]. "This document describes an encoding for the DCMES in XML subject to these restrictions:

- The Dublin Core elements described in the DCMES V1.1 reference can be used
- No other elements can be used
- No element qualifiers can be used
- The resulting XML cannot be embedded in web pages

Then XML schema for "Simple Dublin Core" in 2002 [DCXMLS]: "This schema defines terms for Simple Dublin Core, i.e. the 15 elements from the http://purl.org/dc/elements/1.1/ namespace, with no use of encoding schemes or element refinements." The guidance document "Using Dublin Core" defined "Simple Dublin Core" as: "The fifteen Dublin Core elements used without qualifiers, that is without element refinement or encoding schemes. Sometimes referred to as Dublin Core simple" [GLOSSARY].

Why do we need an AP to say what Simple DC is? Because you have to say the 15 properties are optional and repeatable. New way of seeing things: "Simple DC is a description set with one description that describes a resource with 15 optional property usages." Cite ABSTRACT-MODEL and DSP-MODEL.

This profile exemplifies several aspects of the DCMI model:

- Literal/Non-Literal

- What is specified in a profile and what can remain unconstrained.
- Difference between vocabulary and profile.
- Domains and ranges
- Description template (description sets can contain multiple descriptions)

Note: Simple DC includes value string language, and this is optional. Documentation mentioning "Simple Dublin Core" should be revised to point to this Simple DC AP.

Issues:

- Proposed range of dcterms:description is rdfs:Resource -- see discussion of http://dublincore.org/documents/2007/07/02/domain-range/ in http://dublincore.org/usageboardwiki/RangesIssues

Description template

Title

Property	http://purl.org/dc/terms/title
Literal?	Yes
Definition	A name given to the resource.
Value (Literal)	**Syntax Encoding Scheme:** **Occurrence** disallowed **Language Constraint:** **Occurrence** optional

Creator

Property	http://purl.org/dc/terms/creator
Literal?	No
Definition	An entity primarily responsible for making the resource.
Comment	Examples of a Creator include a person, an organization, or a service. Typically, the name of a Creator should be used to indicate the entity.

Value (Non-Literal)	**Value URI Constraint**: **Occurrence** disallowed **Vocabulary Encoding Scheme Constraint** **Occurrence**: disallowed **Value String Constraint:** Max occurrence 1 *Syntax Encoding Syntax Constraint*: **Occurrence**: disallowed *Language Constraint*: **Occurrence** optional

Subject

Property	http://purl.org/dc/terms/subject
Literal?	No
Definition	The topic of the resource.
Comment	Typically, the subject will be represented using keywords, key phrases, or classification codes. Recommended best practice is to use a controlled vocabulary. To describe the spatial or temporal topic of the resource, use the Coverage element.
Value (Non-Literal)	**Value URI Constraint**: **Occurrence** disallowed **Vocabulary Encoding Scheme Constraint** **Occurrence**: disallowed **Value String Constraint:** Max occurrence 1 *Syntax Encoding Syntax Constraint*: **Occurrence**: disallowed *Language Constraint*: **Occurrence** optional

Description

Property	http://purl.org/dc/terms/description
Literal?	Yes

Definition	An account of the resource.
Comment	Description may include but is not limited to: an abstract, a table of contents, a graphical representation, or a free-text account of the resource.
Value (Literal)	**Syntax Encoding Scheme:** **Occurrence** disallowed **Language Constraint:** **Occurrence** optional

Publisher

Property	http://purl.org/dc/terms/publisher
Literal?	No
Definition	An entity responsible for making the resource available.
Comment	Examples of a Publisher include a person, an organization, or a service. Typically, the name of a Publisher should be used to indicate the entity.
Value (Non-Literal)	**Value URI Constraint**: **Occurrence** disallowed **Vocabulary Encoding Scheme Constraint** **Occurrence**: disallowed **Value String Constraint:** Max occurrence 1 *Syntax Encoding Syntax Constraint*: **Occurrence**: disallowed *Language Constraint*: **Occurrence** optional

Contributor

Property	http://purl.org/dc/terms/contributor
Literal?	No
Definition	An entity primarily responsible for making contributions to the resource.
Comment	Examples of a Contributor include a person, an organization, or a service. Typically, the name of a Contributor should be used to indicate the entity.

Value (Non-Literal)	**Value URI Constraint**: **Occurrence** disallowed **Vocabulary Encoding Scheme Constraint** **Occurrence**: disallowed **Value String Constraint:** Max occurrence 1 *Syntax Encoding Syntax Constraint*: **Occurrence**: disallowed *Language Constraint*: **Occurrence** optional

Date

Property	http://purl.org/dc/terms/date
Literal?	Yes
Definition	A point or period of time associated with an event in the lifecycle of the resource.
Comment	Date may be used to express temporal information at any level of granularity. Recommended best practice is to use an encoding scheme, such as the W3CDTF profile of ISO 8601 [W3CDTF].
Value (Literal)	**Syntax Encoding Scheme:** **Occurrence** disallowed **Language Constraint:** **Occurrence** optional

Type

Property	http://purl.org/dc/terms/type
Literal?	No
Definition	The nature or genre of the resource.
Comment	Recommended best practice is to use a controlled vocabulary such as the the DCMI Type Vocabulary [DCTYPE]. To describe the file format, physical medium, or dimensions of the resource, use the Format element.

Value (Non-Literal)	**Value URI Constraint**: **Occurrence** disallowed **Vocabulary Encoding Scheme Constraint** **Occurrence**: disallowed **Value String Constraint:** Max occurrence 1 *Syntax Encoding Syntax Constraint*: **Occurrence**: disallowed *Language Constraint*: **Occurrence** optional

Format

Property	http://purl.org/dc/terms/format
Literal?	No
Definition	The file format, physical medium, or dimensions of the resource.
Comment	Examples of dimensions include size and duration. Recommended best practice is to use a controlled vocabulary such as the list of Internet Media Types [MIME].
Value (Non-Literal)	**Value URI Constraint**: **Occurrence** disallowed **Vocabulary Encoding Scheme Constraint** **Occurrence**: disallowed **Value String Constraint:** Max occurrence 1 *Syntax Encoding Syntax Constraint*: **Occurrence**: disallowed *Language Constraint*: **Occurrence** optional

Identifier

Property	http://purl.org/dc/terms/identifier
Literal?	No

Definition	An unambiguous reference to the resource within a given context.
Comment	Recommended best practice is to identify the resource by means of a string conforming to a formal identification system.
Value (Non-Literal)	**Value URI Constraint**: **Occurrence** disallowed **Vocabulary Encoding Scheme Constraint** **Occurrence**: disallowed **Value String Constraint:** Max occurrence 1 *Syntax Encoding Syntax Constraint*: **Occurrence**: disallowed *Language Constraint*: **Occurrence** optional

Source

Property	http://purl.org/dc/terms/source
Literal?	No
Definition	A related resource from which the described resource is derived.
Comment	The described resource may be derived from the related resource in whole or in part. Recommended best practice is to identify the related resource by means of a string conforming to a formal identification system.
Value (Non-Literal)	**Value URI Constraint**: **Occurrence** disallowed **Vocabulary Encoding Scheme Constraint** **Occurrence**: disallowed **Value String Constraint:** Max occurrence 1 *Syntax Encoding Syntax Constraint*: **Occurrence**: disallowed *Language Constraint*: **Occurrence** optional

Language

Property	http://purl.org/dc/terms/language
Literal?	No
Definition	A language of the resource.
Comment	Recommended best practice is to use a controlled vocabulary such as RFC 4646 [RFC4646].
Value (Non-Literal)	**Value URI Constraint**: **Occurrence** disallowed **Vocabulary Encoding Scheme Constraint** **Occurrence**: disallowed **Value String Constraint:** Max occurrence 1 *Syntax Encoding Syntax Constraint*: **Occurrence**: disallowed *Language Constraint*: **Occurrence** optional

Relation

Property	http://purl.org/dc/terms/relation
Literal?	No
Definition	A related resource.
Comment	Recommended best practice is to identify the related resource by means of a string conforming to a formal identification system.
Value (Non-Literal)	**Value URI Constraint**: **Occurrence** disallowed **Vocabulary Encoding Scheme Constraint** **Occurrence**: disallowed **Value String Constraint:** Max occurrence 1 *Syntax Encoding Syntax Constraint*: **Occurrence**: disallowed *Language Constraint*: **Occurrence** optional

Coverage

Property	http://purl.org/dc/terms/coverage
Literal?	No
Definition	The spatial or temporal topic of the resource, the spatial applicability of the resource, or the jurisdiction under which the resource is relevant.
Comment	Spatial topic and spatial applicability may be a named place or a location specified by its geographic coordinates. Temporal period may be a named period, date, or date range. A jurisdiction may be a named administrative entity or a geographic place to which the resource applies. Recommended best practice is to use a controlled vocabulary such as the Thesaurus of Geographic Names [TGN]. Where appropriate, named places or time periods can be used in preference to numeric identifiers such as sets of coordinates or date ranges.
Value (Non-Literal)	**Value URI Constraint**: **Occurrence** disallowed **Vocabulary Encoding Scheme Constraint** **Occurrence**: disallowed **Value String Constraint:** Max occurrence 1 *Syntax Encoding Syntax Constraint*: **Occurrence**: disallowed *Language Constraint*: **Occurrence** optional

Rights

Property	http://purl.org/dc/terms/rights
Literal?	No
Definition	Information about rights held in and over the resource.
Comment	Typically, rights information includes a statement about various property rights associated with the resource, including intellectual property rights.

Value (Non-Literal)	**Value URI Constraint**: **Occurrence** disallowed **Vocabulary Encoding Scheme Constraint** **Occurrence**: disallowed **Value String Constraint:** Max occurrence 1 *Syntax Encoding Syntax Constraint*: **Occurrence**: disallowed *Language Constraint*: **Occurrence** optional

References

[RFC2413] Stuart Weibel, John Kunze, Carl Lagoze, Misha Wolf, Dublin Core Metadata for Resource Discovery [RFC 2413], IETF, September 1998, http://www.ietf.org/rfc/rfc2413.txt.

[DCQUALIFIERS] http://www.dublincore.org/documents/2000/07/11/dcmes-qualifiers/

[DCMES-XML2] http://dublincore.org/documents/2000/11/dcmes-xml/

[DCMES-XML1] http://dublincore.org/documents/2000/07/14/dcmes-xml/

[DCMES-XML3] http://dublincore.org/documents/2001/09/20/dcmes-xml/

[DCXMLS] http://dublincore.org/schemas/xmls/

[GLOSSARY] http://dublincore.org/documents/2003/08/26/usageguide/glossary.shtml

[DSP] http://dublincore.org/architecturewiki/DescriptionSetProfile

[ABSTRACT-MODEL] http://dublincore.org/documents/2007/06/04/abstract-model/

SimpleDublinCore (last edited 2007-08-16 11:42:54 by MikaelNilsson)