Proposal to remove the three MODS elements from DC-Lib
Robina Clayphan, 14 September 2006

The Usage Board (UB) of the Dublin Core Metadata Initiative (DCMI) will review DC application profiles (DCAPs) and register them if they conform to the DC Abstract Model (DCAM) and meet the other criteria set out in the UB Process document at http://dublincore.org/usage/documents/process/#section2-2. Such profiles are seen primarily "as a form of documentation, the purpose of which is to help implementer communities harmonize their metadata practice. In the longer term, machine-processable versions of such APs based on data models such as RDF will provide a basis for automating metadata interoperability functions such as semantic crosswalks and format conversions."

The DC Libraries application profile (DC-Lib) has been in existence for several years now but cannot progress through the review and registration processes due principally to the incorporation of three elements from the MODS namespace: dateCaptured, edition and location. The reasons for this are explained in the later sections of this paper.

Proposal
To move on from the current impasse it is proposed:

1. to remove the three problem elements immediately. This can be achieved fairly simply by a poll amongst the members of the DC Libraries working group. A discussion of this will be held during the annual meeting on 5 October 2006 in Colima, Mexico followed by a vote. The poll will then be extended to the wider DC Libraries list.

2. to start the process of finding replacement terms from other namespaces. There is already a candidate for "location" as a similar term has been established for the Collection Description application profile. A proposal to adopt this will be put to the WG meeting for voting and extended to the wider list. The WG will need to undertake the task of identifying alternatives for the other two terms - dateCaptured and edition.

3. if alternative terms cannot be found the WG will need to undertake the task of drawing up proposals to the Usage Board for the creation of new DCMI terms.

Background

The September 2002 version of the DC Libraries Application Profile (DC-Lib) was updated to incorporate three terms taken from the MODS namespace; dateCaptured, edition and location. The terms had originally been proposed to the Usage Board as new elements or element refinements for inclusion in the DCMI namespace. The 2002 version of DC-Lib can be seen at:
http://dublincore.org/documents/2002/09/24/library-application-profile/

1. The term “captured” was proposed as a refinement of the DC element “date”. The text of the proposal can be seen at http://dublincore.org/usage/meetings/2002/05/captured-date_prop.html

   The UB decision was that the element "dateCaptured" from the MODS namespace should be used instead, the main reason being that the term was already available in that namespace. The full decision can be seen at http://dublincore.org/usage/decisions/2002/2002-02.captured.shtml

2. The term “version” was proposed as a refinement of the DC element “description”. The text of the proposal can be seen at http://dublincore.org/usage/meetings/2002/05/description-version_prop.html

   The UB recommended using the element “edition” from the MODS namespace as the term was already defined there with the same semantics. The full decision can be seen at http://dublincore.org/usage/decisions/2002/2002-02.version.shtml

3. The term “holdingLocation” was proposed as a new DC element. The text of the proposal can be seen at http://dublincore.org/usage/meetings/2002/05/holding-location_prop.html

   The UB recommended using the element “location” (now physicalLocation) from the MODS namespace as the term was already defined there with the same semantics. The full decision can be seen at http://dublincore.org/usage/decisions/2002/2002-02.holdingLocation.shtml

These decisions were in line with the principles of interoperability and the re-use of metadata terms in the development of DC application profiles (DCAPs). It was felt that it was not an appropriate role for DCMI to be the guardian of an ever-expanding set of terms, especially as many terms were defined and maintained in other communities and could therefore be re-used in DCAPs. Since 2002, however, the understanding of data models has matured considerably. The way relationships between resources and their properties are structured has raised significant questions about the feasibility of simply mixing and matching elements based on little more than semantic equivalence.

Issues that have arisen

The difficulty with reusing the MODS terms in the recommended way did not become fully apparent until a draft XML schema for DC-Lib was produced in 2005. http://epub.mimas.ac.uk/DC/dc-lib/xsd/dclib.xsd

URIs and referencing the elements

In the MODS schema all three terms are sub-elements of a higher level container element: dateCaptured and edition are below originInfo, and physicalLocation (the name of the equivalent element in MODS) is below the container term Location. The immediate, and

more easily understood, problem with this situation is that the desired elements cannot be directly referenced, firstly because they are not at the top level and secondly because they do not each have a unique and persistent URI. Not being at the top level means creating such a URI would be problematic but not necessarily impossible. There is a school of thought that feels that whilst remaining within the MODS context the elements could be globalised and the necessary URIs could be created. This is as yet unproven and, even if it could be done the second and more profound issue outlined below would seem to make it a redundant exercise.

Underlying Models

The more intractable problem, is that MODS "elements" and DC "elements" are simply not the same kind of thing and cannot therefore be mixed and matched as if they were. It is true that those elements in question share the same semantics, but that is as far as the similarity goes. The main, and seemingly insuperable, problem is that the two set of terms do not share the same underlying conceptual model.

MODS elements are components in a hierarchical data structure and their interpretation is defined in terms of that structure. MODS elements are all containers of one sort or another in a tree data structure, some elements exist only to contain other elements, some are contained in those higher level elements (sub-elements); they can have attributes; at the end of a branch some elements can contain a piece of data about the resource being described. The interpretation of a MODS element depends on its position in the structure. An example is the "extent" element which appears twice in the MODS schema, once within the "physicalDescription" container element and once as a sub-element of the "part" container element – it follows that it makes no sense to talk about the meaning of the mods:extent element in isolation as its meaning can only be distinguished in the context of the structure of the MODS schema.

For the DC elements, more accurately called "properties", the DCAM specifies how to use them (and other types of term) to make statements about the relationship between resources. One of the resources will be the subject of a set of statements that constitute the description of that resource. A small example of this: the human resource "Robina Clayphan" is the "dc:creator" of the textual resource "this paper about MODS and DC" – there is a creator relationship between the first named resource and the second. This corresponds with the RDF model of a set of triples (resource, property, value). DCMI elements are not therefore containers in the MODS sense, they are properties that indicate the types of relationship that exist between two resources. This can be seen very clearly in the recently proposed XML binding for DC descriptions (http://dublincore.org/documents/2006/05/29/dc-xml/) where the URI for a particular resource is followed by a series URIs for the DC terms being used, each with an associated value string or URI for another resource.

Concepts like "sub-element" and "child element" make perfect sense in the MODS hierarchical model but are meaningless in the DCAM where all elements are equal; and conversely, notions such as element refinement which work in the DCAM and RDF models have no place in the MODS model.

To summarise: the MODS schema defines a structure of containers into which content (typically in the form of a text string) can be put, whereas the DC metadata set is a set of terms defining specific types of relationship between two resources. A DCAP has to be based on a single underlying model – by definition this must be the DCAM. It cannot be expressed in XML unless it is based on a single model. Any mixing and matching has to take place within the context of that model and the kind of hybridisation implied by including MODS elements cannot work.

The full discussion of the MODS and DC models and these other issues can be found in the February 2005 and April 2006 archives of the DC Libraries list.

URIs, Qnames etc.

A certain degree of confusion has been caused by the way elements and URIs etc. are commonly written down. We are all familiar with the shorthand way of referring to, for example mods:name or dc:date, but what these names conceal is what the element being referred to actually is. There is a discussion about URIs, namespaces, XML Qnames, XML elements etc in the April 2006 archive of the DC Libraries working group which is not reprised here due to the less than perfect understanding on the part of the author.

Acknowledgements

My grateful thanks to those who participated in the discussion of these issues and especially to Ann Apps and Pete Johnston for their knowledgeable and patient assistance.