IFLA
2005
OSLO

World Library and Information Congress:
71th IFLA General Conference and Council

"Libraries - A voyage of discovery"

August 14th - 18th 2005, Oslo, Norway

Conference Programme:
http://www.ifla.org/IV/ifla71/Programme.htm

August 6, 2005

Code Number: 175-E121
Meeting: 121 UNIMARC with Information Technology

MARCXML Sampler

Sally H. McCallum
Library of Congress
Washington, DC, USA

Abstract

At the IFLA conference in Glasgow, three years ago, the Information Technology Section organized a workshop on metadata. At that workshop MARCXML was presented, along with plans and expectations for its use. This paper is an update to that report. It reviews the development of an XML schema for MARC 21 and the MARCXML tool kit of transformations. The close relationship of MARCXML to the recent ISO standards work associated with MARC in XML is described. Sketches of interesting applications follow with uses that range from MARCXML as a switching format to a maintenance tool to a record communication format for new XML-based protocols.

Background

The development of an XML version of MARC 21 was critical for the format. The economically deep commitment to MARC data elements, proliferation of schemas beyond the library community control, and the rapidly growing XML tool environment mandated an evolutionary path into XML for MARC 21. At the Library of Congress this was interpreted to mean taking advantage of XML by first establishing a standard MARC 21 in an XML structure; next developing a compatible but simpler companion to MARC 21 in XML (which became the Metadata Object Description Schema (MODS)); and then developing a coordinated set of tools for record transformations. The products of these activities would provide flexible transition options for the future and hopefully avoid having many different but not quite the same XML schemas for MARC 21.

In the mid 1990s, when SGML, the precursor of XML, was the current tool for experimentation, the Library of Congress developed an SGML DTD for MARC 21. It had some interesting features, including a separately defined tag to identify every data element in MARC. This enabled detailed validation and enabled elements to be used in isolation and still be fully identified. MARC 21 in SGML was packaged into two DTDs: a Bibliographic DTD that included all of the data elements for the MARC 21 Bibliographic, Holdings, and Community Information formats; and an Authority DTD that contained the data elements for the MARC 21 Authority and Classification formats. This method yielded very large DTDs, since SGML (and XML) are naturally verbose, and the tagging approach mandated a DTD element specification for every MARC subfield or coded character position. The record instances, however, were not overly large.

Based on the experience with the SGML DTDs, the Library of Congress created an XML schema for MARC 21 in the early 2000s. (At the same time, the SGML DTDs were converted to XML DTDs and are still keep available from the MARC 21 web site. Several users have stated that they find them appropriate for certain applications, especially those needing extensive validation of records.) The new XML schema, called MARCXML, was designed with several differences from the DTDs, which made it a very short standard, so it was nicknamed Aslim@. A key characteristic of MARCXML is that it produces an exact equivalent of the MARC 21 (2709) record so that roundtrip conversion to and from it is lossless. This schema has been widely used and is the basis for the international standard for an XML version of the MARC structure that Danish Standards has proposed to ISO (described later). Some of the characteristics of the MARCXML schema are the following.

(1) The highest level elements are <collection> and <record>, to enable a group of records to be assembled into a package and to clearly define a single record.

(2) Field tags and indicators that are found in MARC 21 are treated as attributes. This allowed the first level number of elements to be radically reduced in MARCXML. Only 3 basic format-related elements are needed: <leader>, <controlfield>, and <datafield>. The simple and flexible approach of specifying any three digit string as a possible MARC tag gave up the ability to carry out some types of validation with the Aslim@ schema that were possible with the earlier detailed DTD. Instead, the accompanying architecture (see below) called for a validation tool outside the schema for applications where that was needed.

Example:

```
<datafield tag="245" ind1="1" ind2="0">
```

(3) A <datafield> has one child element, <subfield> (which is repeatable for each subfield), and <subfield> has an attribute for specifying the subfield code.

Example:

```
<datafield tag="082" ind1="0" ind2="0">
        <subfield code="a">796.6/4/0943</subfield>
        <subfield code="2">20</subfield>
</datafield>
```

(4) The content of each <controlfield>, MARC tags, 001-009, is treated as a string. This means that for the MARC 008, for which coded elements are defined by character position, the full 40 bytes of the 008 are transformed to XML including blanks. Fortunately XML has a simple schema specification that keeps blanks from being compressed or otherwise tampered with, essential for treating some control fields as strings.

Example:

```
<controlfield tag="008">931129s1994    wauab        001 0 eng   </controlfield>
```

(5) The <leader> is likewise treated as a string. While some of the information in the leader is only relevant to an ISO 2709 record, it is simply carried over where it is easy to discard in subsequent transformations.

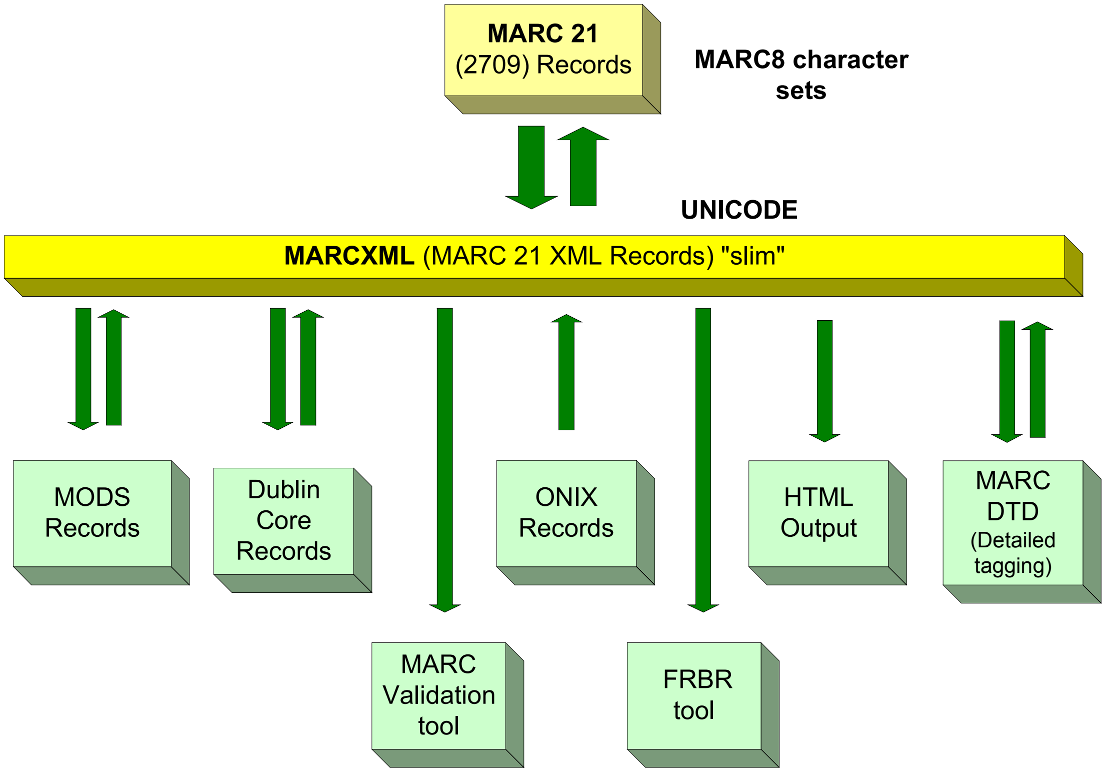
Example:

```
<leader>00637cam  2200193 a 4500</leader>
```

MARC Tool Kit

As the architecture presented in 2002 indicated, MARCXML can then be used for data exchange or as a Abus@ for further transformations or processes. The architecture and the downloadable transformations are called the AMARC Tool Kit@ and can be accessed on the MARC 21 web site (www.loc.gov/marcxml). The illustration of the tool kit uses boxes to indicate the various data formats or functions, and arrows to indicate the transformations.

MARC 21 Tool Kit for the XML Environment



An important part of the tool kit is the provision of converters for transforming data from MARC 21 (2709) to MARCXML and back, including character set conversion to and from Unicode. These converters can be downloaded from the MARC website and used by others in their own systems where they can also shape them to their own data and needs. The conversion software between MARC 21 (2709) and MARCXML is in part adapted from an extensive set of programs for manipulating MARC 21 data developed by Bas Peters in the Netherlands and made available by him as open source software.

The MARC tool kit includes several other interesting transformations that are downloadable. These include:

$ format transformations to DC (several flavors) and MODS
$ format transformations from DC, MODS, ONIX, and oai_marc
$ MARCXML to FRBR display tool
$ MARCXML record validation tool
$ transformations to and from the earlier MARC21 XML DTD

The Library of Congress sees these transformations provided from the MARC 21 maintenance agency as being valuable to the community to help maintain the savings and interoperability built up through use of a common format. They have also served as a starting point or stimulus to users and innovators.

The ISO standard

In 2004, Danish Standards introduced into ISO an XML schema, called MarcXchange, that has approximately the same relationship to MARCXML as ISO 2709 has to MARC 21. Since the MARCXML schema is defined in a broad and general way, it could be used with little change for MarcXchange. The family of bibliographic formats which follow ISO 2709 and generally have MARC somewhere in their name have been consistently implemented with the following selection of ISO 2709 options and MARCXML specifies those limits: only 2 indicators per field, 1 character subfield codes. ISO 2709 is more liberal, however, and allows up to 9 indicators per field and up to 9 characters for subfield codes, so the MarcXchange had to broaden that specification to allow the additional indicators and subfield codes. MarcXchange also makes a few actual extensions to the ISO 2709 specifications. The already widely used MARCXML remains a valid implementation of MarcXchange, however. The main changes from ISO 2709 are the following.

1) Addition of an attribute, Aformat@, to the high level element <record> that identifies the specific MARC content designation (tagging, subfield codes, coded data) used in the record. Examples of values for this attribute might be MARC21, Unimarc, danMARC2, and Ibermarc.

2) Addition of another attribute, Atype@, to the element <record> that identifies the genre of record. Values for MARC 21 might be bibliographic, authority, holdings, classification, and community. This is already also an addition to MARCXML.

3) Specification that the 00X tags may be used with the element <datafield>, in addition to their specification under the element <controlfield>. This is an extension of ISO 2709, but does not invalidate any strict use of 2709.

The Danish draft was submitted to the ISO Subcommittee in March 2005 for ballot as a new work item. Since the draft standard was highly developed already, the November 2004 meeting of ISO TC46 SC4 had recommended that the ballot also inquire whether the draft could proceed to Draft International Standard level, thus eliminating delays in finalization of the standard.

The new work item ballot was approved on June 15, 2005 with comments advising that another review of MarcXchange against 2709 be made to assure that 2709 is fully supported. The comments and draft will be reviewed and balloting at the next stage will probably take place by early 2006.

A Sampling of MARCXML Uses

This section describes several specialized applications that use MARCXML. These applications use MARCXML to communicate MARC records and to enable use of the large number of XML tools to manipulate records B frequently to carry out processing that would require more resources with 2709 records.

Metadata switch: OCLC=s terminology services project

An important project of the OCLC Office of Research is one called ATerminology Services@. The goal of this project is to offer web services that are machine-to-machine applications that can be used in various ways. The terminology services project deals with knowledge organization vocabularies, such as authority files, subject heading systems, thesauri, and classification schemes. A web service that provides mappings from a term in one vocabulary to one or more terms in another vocabulary is an example of a terminology service. Microsoft research panes are used to provide dictionary services, and to enable 3rd party extensibility.

At the heart of this work is the metadata switch, a normalizing format for the data from the different databases that are initially in various formats: XML, html, MARC 21, etc. MARCXML is used to normalize the data since MARC 21 is rich is support of details of thesauri and classification systems, making it a logical choice. The MARCXML version of MARC21 is used because it enables the enhancement of the vocabularies with XML utilities and XSLT style sheets, and it is easier to combine non-MARC with MARC and other XML formatted data.

Some of the vocabularies that have been mapped are DDC, MESH, ERIC Thesaurus, Guidelines on Subject Access to Individual Works of Fiction, Drama, Etc. (GSAFD), Newspaper Genre List, LCC, and LCSH. The mapped records are made available in MARCXML and Dublin core via a browser and to machines through OAI-PMH.

Record edit and crosswalk applications

Terry Reese from the University of Oregon in the United States, several years ago developed the versatile, free MARC 21 editing utility, MareEdit. <http://oregonstate.edu/~reeset/marcedit/html/index.html> MarcEdit started out as an easy to use editing tool, along the lines of the MARCMaker and MARCBreaker (DOS-based) tools that the Library of Congress had developed and posted for download over 10 years ago. MarcEdit has their functionality but it is set in the current environment B windows and enabled for integration with other software applications and programming/scripting languages. Reese used MARCXML as a key element of the new MARC 21 editing utility, and added the ability to edit MARCXML and MODS files directly in the tool.

MarcEdit=s rapid take-up by MARC 21 users encouraged further development, and MARCXML is also central to the crosswalk tools now provided by Reese. These include data conversions from Dublin Core, Encoded Archival Description (EAD), and geospatial object metadata (FGDC format) to MARC 21. For example, the crosswalk tools make it simple to take DSpace metadata created by faculty and staff in a local DSpace repository and convert it to MARC 21 via MARCXML for integration into the library catalog or other databases.

Record manipulation and maintenance tool

Bill Jones of New York University (NYU) reports extensive use of MARCXML to carry out specialized and routine tasks on the University=s very large MARC 21 catalog. Using selection devices to obtain subsets of records, he brings the records out into MARCXML where he can use XSL transformations on thousands of records at a time. It is akin to a well-controlled global update, with great flexibility. Not only does he adjust tagging and add/delete fields and subfields, but he also changes the data content of fields.

Some of his recent projects included the following. MARC 21 records are obtained by NYU from multiple sources. Before load, they are converted to MARCXML where they are corrected, items deleted, search information added, and electronic data links adjusted. For example, for electronic resources, the linking URIs need to be adapted to the local situation B and moved from the bibliographic record to a newly created MARC 21 holdings record where NYU stores that information. The incoming MARC 21 bibliographic records are converted to MARCXML where the changes take place and the revised bibliographic and new holdings records are then converted from MARCXML to MARC 21 (2709) and loaded to the catalog. In another project, to improve the consistency of reproduction notes before load, records are often processed in MARCXML to fill in missing data from information in the record and system supplied data.

A common NYU application is to use XML tools to process batches of records from the catalog into XML for identification of special groups of records. For example, when the digital unit at NYU completed a project to digitize a number of sound recordings, they supplied Jones with the recording publisher number and the URI link to the recording. All of the existing sound recording records in the catalog were transformed into MARCXML, the records with matching publisher numbers identified, and the URIs inserted. After conversion back to MARC 21, they were reloaded to the catalog. These extraction capabilities in the XML environment were also applied to a cooperative project whose partner needed all the records with certain country of publication, language, and genre characteristics.

XML-based protocols

MARCXML is widely used for exchange of MARC 21 data in new protocols that were developed and operate in the XML environment, such as the Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH) and the SRU/SRW information retrieval protocol. When MARCXML was finalized, the Open Archives Initiative announced that it was changing its MARC recommendation from a project-specific MARC schema (oai_marc) to the MARCXML schema. While the two syntaxes are similar, the OAI preferred to use an externally maintained standard for the MARC record rather than maintaining a syntax themselves. For convenience, the MARC tool kit contains an XSL style sheet that converts oai_marc data to MARCXML. Thus today a number of OAI data provider sites have MARC 21 records that they expose in MARCXML in addition to the mandatory DC. The Library of Congress, for example, exposes its metadata for digital objects in its American Memory collection for OAI harvesting in MARCXML, MODS, and DC.

The Networked Digital Library of Theses and Dissertations (NDLTD), a global project administered by staff at Virginia Tech in the United States, uses the OAI protocol to gather and expose metadata about electronic theses. While the critical focus of that project is to encourage the creation and use of electronic theses, this is done primarily via metadata. The OAI protocol requires unqualified DC metadata, but also supports use of MARC, in XML form. OCLC, which regularly harvests the metadata from NDLTD, sees a lot of MARCXML. Having a standard format for most of the data is a great help for data harvesters.

The Search/Retrieve URL service and Search/Retrieve Web service protocols also use MARCXML, since they require that records be in XML syntax. MARCXML is thus the retrieval vehicle for searches requesting MARC 21 records in their entirety. The Virtual International Authority File (VIAF), under development at OCLC as a joint project of the Library of Congress, the Deutsche Bibliothek, and OCLC, will be accessible via SRU and OAI. The SRU view of VIAF will be entirely MARCXML metadata. Library of Congress also uses MARCXML in its SRU and SRW implementation. The Library's SRU/SRW software offers to send retrieved records in MARCXML (and MODS and DC), although they are initially exported from the Library=s catalog in MARC 21.

Library of Congress

And finally the Library of Congress is using MARCXML and some of the posted transformations in a number of ways. Staff regularly create MODS records from MARC 21 records, via MARCXML, for our digital activities. Data such as content abstracts and tables of content can easily be moved from ONIX to MARC21 (2709) via MARCXML for addition to our catalog records. The Library of Congress also offers distribution of all of its MARC 21 cataloging records in the MARCXML schema, in addition to the ISO 2709 structure, even though the expectation is that 2709 will be the preferred format structure for a number of years to come.

In Summary

MARCXML provides a basis for evolution while maintaining standardization. The various XML schema and transformations in the MARC Tool Kit have proven useful for XML communication of MARC data, opening MARC 21 to XML programming tools and presentation style sheets, standardizing MARC 21 for OAI harvesting, and standardizing transformations to and from other standard formats such as MODS, DC, and ONIX.