2001

Applying XML to the Bibliographic Description

David J Fiander, *Western University*

Applying XML to the Bibliographic Description

David J. Fiander

ABSTRACT. Over the past few years there has been a significant amount of work in the area of cataloging internet resources, primarily using new metadata standards like the Dublin Core, but there has been little work on applying new data description formats like SGML and XML to traditional cataloging practices. What little work has been done in the area of using SGML and XML for traditional bibliographic description has primarily been based on the concept of converting MARC tagging into XML tagging. I suggest that, rather than attempting to convert existing MARC tagging into a new syntax based on SGML or XML, a more fruitful possibility is to return to the cataloging standards and describe their inherent structure, learning from how MARC has been used successfully in modern OPACs while attempting to avoid MARC's rigid field-based restrictions.

KEYWORDS. XML, AACR, bibliographic description, standards, Web

INTRODUCTION

Fifty years ago the Library of Congress began looking into the possibility of using computers to automate the production of library catalog

David J. Fiander, MLIS, is Reference and Instructional Librarian, Allyn and Betty Taylor Library, University of Western Ontario, London, ON N6A 5B7, Canada.

The author would like to thank Dr. Grant Campbell, Assistant Professor, UWO, for his encouragement.

cards. After a decade of research and experimentation, the MARC data format was finalized and put into production use.[1] Today the MARC format continues to be used as a communication and data exchange format, and all large cataloging systems support importing and exporting MARC format records.

Over the forty years since MARC was first mooted, it has grown and been modified organically as the abilities of library automation systems advanced and as more experience was gained in using MARC to catalog everything that a library encounters. This has led to inconsistencies within the MARC format between different fields that contain the same type of data and, even with all the maintenance effort, some types of data that cannot be encoded in a useful way. For example, Miller notes that there are five different fields for personal names, each with roughly seven subfields and that, even with all that, there is no way to distinguish the forename from the surname in the coding.[2]

In the early 1980s, at the same time that MARC was starting to be used for online catalogs rather than for the production of cards, IBM began working with the International Organization for Standardization (ISO) to produce a standard document markup language based on IBM's GML product. The result of this standardization process was the Standard Generalized Markup Language, SGML.[3] The SGML standard is not, directly, a text formatting standard. It defines a specialized "language" for describing the structure of documents, or any type of textual data, leaving the interpretation of the data to another program. Thus, SGML is ideal for encoding strongly structured textual data for communication between computer systems. Unfortunately, SGML is a large complicated standard; the computing resources necessary to support the complete standard, with all its options, cannot be deployed broadly at this time. This difficulty led to the World Wide Web Consortium's development of the Extensible Markup Language, or XML, which is "a subset of SGML" that "has been designed for ease of implementation and for interoperability with both SGML and HTML."[4] One of the best examples of strongly structured data that librarians will be familiar with is the bibliographic catalog record. While there has been a lot of work into web-based metadata systems (Vellucci provides a survey of the literature and a discussion of the various metadata systems relationships to traditional library authority control issues; and Brugger discusses the difficulties involved in cataloging digital libraries[5]), there has been little work examining the possibility of using XML for complete bibliographic description of traditional library materials.

Although Miller describes several problems with the MARC format,[6] primarily related to consistency and lack of tagging that would be of use to users, the fact that it works, and works as well as it does, indicates that bibliographic data is well suited to being described in a structured fashion for use by computers. This structure is inherent in the underlying bibliographic description standards that we use. The Anglo-American Cataloguing Rules provide enough structure to a bibliographic description that knowledgeable catalog users (or at least catalogers) can even determine the different parts of a catalog record in languages they don't understand (see, for example, the figures in Takawashi's paper about the Nippon Cataloguing Code[7]). While XML provides a framework for describing strongly structured data, and includes facilities for nesting data (such as tagging the title within the title and statement of responsibility area or, using Miller's example, the forename and surname of a personal name[8]), it is a "meta" language: it provides a way to describe the format of the information, not the information itself. Before libraries can begin to use XML for bibliographic description, they must all agree on a common structure (or "Document Type Definition," DTD, in SGML terms) that describes the data format, much like they must now agree on the details of the MARC format.

Thus, before libraries can migrate from the existing MARC systems to new XML-based systems, the question "How should we structure the DTD?" or, in other terms, "On what do we base the design of the DTD?" must be answered. This article will describe three possible approaches to developing a DTD for bibliographic data, each with a different starting point:

1. Begin with the MARC format and transliterate it into XML.
2. Begin with the structure of AACR2 and describe it via XML.
3. Take advantage of the change in technology and incorporate some recent research into the area of cataloging into the underlying descriptive codes before creating an XML structure.

Aside from giving catalogers the opportunity to avoid the various flaws identified with the MARC format, XML, or some other SGML-based format, has the advantage that it is directly processable by the users' computers. Rather than requiring the catalog to translate the result of a user's search from a set of MARC records into formatted HTML, the results produced by an XML-based catalog can be passed directly to the user's client software to be formatted, or parsed for automatic processing, however the user desires. Thus, every user of the sys-

tem can customize his or her view of the catalog: ISBD paragraph format, labelled as many current OPACs provide, braille or audio output, or perhaps left as is to be processed by specialized cataloging tools by the library technical services staff.

TRANSLITERATING MARC INTO XML

The most direct method for moving from today's MARC-based OPACs into an XML-based future is by transliterating the MARC fields and subfields directly into an XML DTD, preserving the structure of the MARC exactly. This approach has the advantages that it will be (relatively) straight-forward to design the DTD, that practicing catalogers will be able to apply their existing knowledge of the MARC tagging directly, with little training, and that developing the tools necessary to convert existing MARC records into XML for use by new, XML-based, OPACs will be as straight-forward as developing the DTD. A system based on this approach to XML markup would also have the immediate advantage that it is not just simple to convert from MARC to XML, but the reverse conversion is also simple, allowing older catalogs to continue to use new records that are created in XML. This would be very important during the (probably long) transition period when there would still be a significant number of MARC-based OPACs.

This approach is the one taken by the Library of Congress MARC DTD project[9] and the Cheshire project.[10] The purpose of the Library of Congress's MARC SGML project was to "create standard SGML Document Type Definitions to support the conversion of cataloging data from the MARC data structure to SGML (and back) without loss of data." That parenthetical comment in the rationale for the project had far more than a parenthetical effect on the resulting DTDs; it led to a DTD that exactly parallels the existing MARC format, with all its advantages and disadvantages. Before beginning its work, The Library of Congress defined five design principles: generality, reversibility, flexibility, user friendliness, and relationship to TEI.[11] Of these five principles, the two that most clearly speak to the structure of the resulting DTD are reversibility and user friendliness. The fact that the basic objective of the project was to create a DTD for bibliographic data that allowed for conversion from MARC to SGML and back again is reflected in the reversibility principle. The MARC DTD background Web page states that:

> The mapping of MARC data elements to corresponding SGML encodings was specifically designed to be reversible, that is, conversion from one structure to the other could be done without loss of the intellectual content or information relating to essential elements of the other record structure. Data elements defined in MARC can be moved to SGML with the MARC tagging and semantics intact.[12]

Thus, the fact that there are five different MARC fields for personal names means that there are five different SGML tags for personal names. In fact, the seven MARC subfields for personal names balloon to almost thirty-five SGML tags, since the subfields in MARC all share the same names (single letters or digits), but in SGML, the DTD uses unique names for all the subfields constructed by appending the MARC subfield tag to the MARC field number (for example, the $a subfield of the MARC field 100 (personal author, main entry) is denoted by the SGML tag "mrcb100-a"). While the library recognizes that "the main advantage to defining generic subfield elements–as MARC does, in effect–is a shorter DTD with less possible elements," the design committee felt that the field-specific subfield tags allowed for "easier validation of elements and manipulation of SGML data without the constant need to determine context."[13] In strict XML systems, where the result of a user's query is XML data that is displayed according to XML style sheets at the user's web browser (rather than MARC being translated into formatted HTML by the server), eliminating the need to determine context may simplify the processing necessary on the user's computer. Overall, the principal advantage of a DTD based on transcribing exactly the structures of MARC into XML is simplicity: it is simple to convert existing MARC records into the corresponding XML, to train catalogers to use the new notation, and to deal with the transition period during which MARC and XML systems will both be common.[14]

McDonough concludes that there is no need to migrate from MARC to SGML in general, since MARC succeeds as a communication format, even if SGML is going to be used internally by future cataloging systems.[15] This is another reason for ensuring that the cataloging data DTD closely parallels the format of MARC: there will be continuing translation between the two formats. But McDonough was writing before XML became a common part of mass-market word processing systems and web browsers. Now that it is possible to transmit XML to the user's desktop, XML must now be considered not just an internal data

format but an end-to-end data communications markup language, perhaps building on the experience of the Z39.50 protocol.

The primary disadvantage of pure MARC-to-SGML systems like the MARC SGML DTD and the Cheshire II project is that they fail to take advantage of XML's facilities for describing shared structures within the data. Many of the MARC fields that hold personal names use the subfield indicators identically, but this must be enforced by the committees responsible for maintenance of the format. When one MARC field is changed, all others that have the same structure must be examined and modified in parallel to ensure that such consistency continues to hold true. This same process of cross-reference between different elements must also occur when the MARC XML DTD is modified to ensure that the sub-elements (i.e., sub-fields) are kept consistent. However, by defining a "personal name" element in a DTD, to be used whenever an authorized personal name is required by the cataloging rules, all the different roles that a personal name plays in a bibliographic record are guaranteed to have exactly the same structure, and updating the format of that one element automatically updates all the roles.[16] This style also makes learning the markup simpler, since there are no differences or special cases to remember: a personal name is always a personal name regardless of whether the name is that of an author or that of the subject of a biography.

DESCRIBING THE STRUCTURE OF AACR2

While MARC was designed to be a general framework for bibliographic data, practical considerations imposed by the limited computing power available in the late 1960s meant that the resulting format didn't parallel the descriptive standards directly. The order in which the information was recorded in the MARC record was altered from the standard description of the cataloging process, the Anglo-American Cataloguing Rules, to simplify processing. Thus the description of Part I of AACR is mixed in with the access points of Part II: the 100-level fields of MARC, which define access points, are followed by the 245 descriptive field, then the 246 access point, and then the 300 description, and so on. The point of this jumble is that, by sorting the fields of the record numerically, then the information needed to produce the output record (or card) appears in an order that allows for linear processing of the input record: main entry, description, alternative title indexes, description, notes, tracings. Computing power is now cheap enough that

it's no longer necessary to make things easy for computers at the expense of people.

An alternative to transcribing MARC directly into XML and preserving this confusion of description and access is to return to the descriptive cataloging standards and create a new markup based on the structure of the descriptive standards, while learning the lessons of MARC. At a minimum, an AACR-based DTD would replace the standard punctuation (AACR's slashes, dot-space-hyphen-spaces, semicolons, and parentheses) with XML tags, but further tagging, and tag attributes, would allow for the automation of much of what currently requires separate fields. For example, parallel title information transcribed into the MARC 245 field must currently be repeated in 246 fields. With suitable tagging in an XML format the parallel titles based on the transcription could be read directly out of that transcription, just as the title proper currently is in MARC. (Parallel titles provided to enhance access, such as spine or alternate titles, would still have to be coded separately, as I discuss below.) The closest to this model that any previous work seems to have come is Miller's work on serials and authority records.[17] His group started with MARC, and simplified it, and grouped certain tags together in a way that approximates the AACR layout (the description and associated notes are grouped, as are the relationships between serials). His work, however, has had a practical basis and has focussed on the medical serials environment. While he has created "personal name" and "corporate name" tags, to address the particular issue he found with MARC, the remainder of the tags are an ad hoc assortment of descriptive English names ("title," "ptitle," "note") and MARC field names ("v362," "v245a," "v245q"). Whether or not MARC is too complicated, as Miller claims, simplifying it, and then using the simplified version as the starting point for the development of a new bibliographic data system will lead to problems. Miller's further work on the XMLMARC project demonstrates that migrating MARC records to a more structured XML environment can succeed; however, the project continues to be based primarily on the structure of the MARC record with only the obvious structuring performed.[18] If MARC is appropriate, then continue using it; if MARC's description of data is appropriate, but changing the notation leads to an enhanced environment for the patrons and the library staff, then use something like the Library of Congress's XML MARC project; if MARC is inappropriate, then don't use it as the starting point for a new project.

The Library of Congress XML MARC project has the advantage that it includes all of the information used by modern OPAC systems for in-

dexing and selecting records for users. A complete XML system would have to ensure that it included all of the same data provided by MARC, in some form, in order to ensure that no functionality is lost for the user. This brings up the issue of the transition from MARC to XML. While the XML MARC project ensures that any transition period will be relatively smooth, since data can be converted between the two encodings relatively easily, converting automatically between MARC and an XML format based on AACR itself (rather than based on MARC) will be more difficult, and human validation of the result of the conversion process may be necessary for many records.

Since MARC records contain all of the necessary bibliographic data, it is possible to create the corresponding XML records from the MARC record, albeit with incomplete XML tagging in cases where the MARC format lacks appropriate semantic tags (such as the personal name forename). The question is, "How much of the work can be done automatically, since there are too many records for people to do all the work?" For the most part, the MARC records should be easily convertible into XML: the 245 field will be copied into the appropriate "title and statement of responsibility" tag, and the MARC subfields will be inserted as appropriate. Because one of the goals of this type of project would be to enhance the XML tagging to include logical structure that MARC doesn't support, that extra tagging would not be automatically possible, or at least not easily achieved. For example, tagging the forename and surname in a personal name field will not be possible, although it might be possible to guess, given the location of punctuation within the MARC record. Parallel title information recorded in the MARC 246 fields might be matched against the 245 field to identify where in the transcription the parallel title occurs. With some luck, the 246 field may be automatically eliminated from the XML output. (Note, however, that some equivalent of the MARC 246 field will always be necessary in the XML DTD, since it holds all the variant titles that might appear on a work. These would be recorded in the "access points" section of the XML record.) By reorganizing the bibliographic record into the description followed by the access points, as described by AACR, the cataloger's work flow has changed (back to a pre-MARC work flow, perhaps). Today, while cataloging in a MARC system, the cataloger tends to work in numeric field order, which requires a determination of main entry almost immediately. With an AACR-or ISBD-based tagging system, the cataloger works through the description in its entirety and then moves to providing the access points, and designating the main entry point from that set. Such a system would benefit from reorganizing

Part I of AACR into giving the rules for each of the ISBD areas, rather than by type of material as it now is, before beginning the work of developing the DTD.[19]

STARTING AFRESH

Over the last five years there have been several papers written that look at descriptive cataloging and considered whether it might be feasible to create a more recursive descriptive format that eases the creation of analytic catalog entries.[20] Several researchers have also looked at the related possibility of more clearly demarcating the boundary between "the work" as an abstract intellectual concept and "the item" as a physical manifestation of the intellectual content of the work and creating separate records for each of these that are linked together.[21] Less dramatically, the concept of the "main entry" as a distinguished access point has come under fire in recent years, since all access points provide the same, complete, bibliographic record in online catalogs.[22]

Incorporating these ideas into MARC would be difficult. MARC is, because of its structure, a very "flat" data format: the only levels of description possible are fields and subfields. While it is possible to create "recursive" data structures by creating fields that link two MARC records together into some sort of bibliographic relationship, that approach is clumsy, and there will be problems related to catalog maintenance in the long run. As for eliminating the concept of "main entry" from the format, that may be achieved, at least partly, simply by not using 100-level fields. By using the corresponding 700-level field for any names associated with a work, there is no "main entry" associated with the record. Winke notes that since no modern OPAC allows the user to search just the main entries, that there's no visible difference between a record that has a 100 field and one that doesn't, as long as the name appears in one of the "author" fields.[23] In fact, paragraph 0.5 of the Anglo-American Cataloguing Rules recognizes that some libraries have eliminated the main entry from their catalogs, and notes that the possibility of removing it from the code was considered.[24] But this change in the way that the MARC format is used, to support a basic change in the cataloging codes, seems to be papering over the issue, leaving the hole underneath unfixed at a basic level.

A time of change in the technology is a good time to consider reworking the cataloging codes on which the new technology will be built. By redoing the basic codes, using the work of the IFLA study group on the

functional requirements for bibliographic records, incorporating Yee's work on the concept of the work, and then creating a new coding system based on this new theoretical framework, a dramatic new vision of the catalog is possible.[25] Rather than creating bibliographic and authority records as distinct entities that are cross-linked, it may be possible to create a "mesh" of bibliographic data: creators, or entities; works; and manifestations, at least. Creators are responsible for works, which are manifested in physical (or digital) units.[26] The data currently stored in a single MARC record could be decomposed into the work data, the creator data, and the manifestation data, and then relinked together. With a suitable advanced user interface, the library patrons could walk the mesh of records to find the relationships between the different types of data.

This is clearly not feasible at this time, given the fact that converting the existing MARC databases into this new, deconstructed, format will be almost impossible to achieve with any reasonable level of automation. There is also a great deal of research necessary, into data structures, database implementations, and user interfaces, before work could even begin on such a prototype, but this is the type of thing that is possible with a more flexible, open, data definition architecture like XML. Miller described MARC as a "self-imposed handicap"; that is too strong, but MARC does limit the possibility for discourse about how a future catalog might be structured, because it is such a strict format into which to try to fit the ideas that might be possible with the greater computing power and electronic storage space now available to libraries. Miller also, however, points out that the historic emphasis on description is becoming less and less relevant for the bulk of the materials, both physical and electronic, that libraries are collecting, and refocussing our efforts on providing access over accurate description may be appropriate.[27]

CONCLUSIONS

The rigidity and internal irregularities of MARC are beginning to cause problems for catalogers and users, and MARC is beginning to lag behind current research into bibliographic description standards. Rather than trying to patch up MARC, perhaps it's time to start looking for alternative data formats that provide flexibility for the next forty years. However, before libraries can migrate away from MARC to a newer data format for bibliographic data, much further research into embedding current thought into practical data structures is required. Current Internet metadata proposals are incomplete, in that they only describe Web re-

sources, without adequately describing all of the different types of resources to which a library must provide access; and the proposals are also lacking in the (authority) control that is the "added value" that libraries bring to their collections. The way to move forward is to base new work on the descriptive standards that have been developed by the profession over the last century, ensure that there is a strong grounding in bibliographic concepts, and embed such work in data structuring formats that give libraries the flexibility they need to expand and experiment.

REFERENCES

1. Lois Mai Chan, Cataloging and Classification: An Introduction (New York: McGraw-Hill, 1994), p. 403-412.

2. Dick R. Miller et al., "Restructuring serial, circulation, and traditional bibliographic data for deployment in changing digital environments," presented at the Medical Library Association annual meeting, Chicago, 1999. Available on the web at http://xmlmarc.stanford.edu/Speech.htm.

3. Charles F. Goldfarb, The SGML Handbook (Oxford: Clarendon, 1990).

4. Tim Bray, Jean Paoli, and C.M. Sperberg-McQueen, eds, Extensible Markup Language (XML) 1.0 (World Wide Web Consortium). Available on the web at http://www.w3.org/TR/REC-xml.

5. Sherry L. Vellucci, "Metadata and authority control," Library Resources and Technical Services 44, no. 1 (January 2000): 33-43; Judith M. Brugger, "Cataloging for digital libraries," Cataloging & Classification Quarterly 22 no. 3/4 (1996): 59-73.

6. Miller, "Restructuring."

7. Tadayoshi Takawashi and Yasuo Iwashita, "The concept of a bibliographic unit introduced into the newly revised edition of Nippon Cataloging Rules, 1987 edition and the resultant cataloging object," Cataloging & Classification Quarterly 23, no. 2 (1996): 17-39.

8. Miller, "Restructuring."

9. Library of Congress, "MARC SGML."

10. Jerome P. McDonough, "SGML and the USMARC standard: Applying markup to bibliographic data," Technical Services Quarterly 15 no. 3 (1998): 21-33.

11. The Text Encoding Initiative (TEI) project is providing important texts from the Humanities in a standard markup form that includes bibliographic data that is distinct from the types of bibliographic data used by libraries, but is important for textual research in the Humanities. For information about this project, see http://www.uic.edu/orgs/tei/.

12. Library of Congress, "MARC SGML"

13. Library of Congress, "MARC SGML." See the section "Name and scope of subfield-level elements."

14. McDonough, "SGML and USMARC."

15. McDonough, "SGML and USMARC."

16. Whether the role is indicated by an attribute of the "personal name" element as Miller does in his "Choice" presentation or as a separate element which holds a "personal name" element is a design issue. I prefer the latter solution.

17. Miller, "Restructuring."

18. Dick R. Miller, "XML and MARC: a choice or a replacement?" presented at the ALA Joint MARBI/CC:DA Meeting, Chicago, 2000. Available on the web at http://xmlmarc.stanford.edu/ALA_2000.htm.

19. Tom Delsey, "The logical structure of the Anglo-American Cataloguing Rules–Part I." Drafted for the Joint Steering Committee for Revision of AACR, August 1998. Available on the web at http://www.nlc-bnc.ca/jsc/aacrdel.htm.

20. Takawashi, "Concept."

21. Delsey, "Logical structure"; Martha M. Yee, "What is a work?" In The Principles and Future of AACR: Proceedings of the International Conference on the Principles and Future Development of AACR (Toronto: American Library Association, 1998), 62-104.

22. F.H. Ayres, "Bibliographic control at the crossroads," Cataloging and Classification Quarterly, 20, no. 3 (1995): 5-18; R. Conrad Winke, "Discarding the main entry in an online cataloging environment," Cataloging & Classification Quarterly, 16, no. 1 (1992): 53-70.

23. Winke, "Discarding."

24. Michael Gorman and Paul W. Winkler, eds, Anglo-American Cataloguing Rules, 2nd ed, revised (Chicago: American Library Association, 1988).

25. IFLA Study Group on Functional Requirements for Bibliographic Records, Functional Requirements for Bibliographic Records, UBCIM Publications, n.s. vol. 19 (Munich: K.G. Saur, 1998); Yee, "What is a work?" The IFLA paper is also available on the web at http://www.ifla.org/VII/s13/frbr/frbr.pdf.

26. Delsey, "Logical structure"; IFLA, "Functional requirements."

27. Miller, "Choice."

Received: August, 2001
Revised: December, 2001
Accepted: December, 2001