Serviços de Informação Bibliográfica.
Edifício Portas de S. Miguel, Rua Adriano Lucas, 2º B, 3020-430 Coimbra, Portugal
Telefone: (351) 239 496 162 – Telefax (351) 239 496 162 – N.º Contribuinte: 505007924
http://www.bookmarc.pt – Email: info@BookMARC.pt

An XML representation of the UNIMARC Manual:

A working prototype

Joaquim Carvalho (University of Coimbra, Portugal, joaquim@dei.uc.pt)

Abstract

The UNIMARC manual defines a standard for the formal representation of bibliographic information. For that purpose the UNIMARC manual contains different types of information: structural rules that define that records are composed of a leader, a set of control fields and a set of data fields, with certain syntactic characteristics; content rules, that define required fields and acceptable values for various components of the record; and, finally, examples, explanatory notes, cross references to other points of the manual.

Much of this information must find its way into computer systems where it will be used to validate records, produce indexes, adequately format records for display and, in some cases, provide human readable help.

Providing the UNIMARC manual in XML greatly simplifies the full implementation of the format in computer systems. Our goal was to produce a formal representation of the UNIMARC format, so that the standard can be incorporated in software systems in a transparent way. The outcome is an XML representation of the UNIMARC manual, which can be processed automatically by applications that need to enforce the format rules, provide help information, or vocabularies. We developed a scheme for the UNIMARC manual and a set of software tools that demonstrate its usage.

Introduction

With the adoption of the MARCXML (slim) format by the Library of Congress, in 2002, the usage of XML for representing bibliographic records became mainstream. By adopting a "slim" style, suitable for easy transportation of records between systems, the new schema departed from earlier standards that aimed at a self-validating way of expressing bibliographic records. By "self-validating" we mean a format that would only allow the construction of valid records. It is now accepted the argument for the separation between "transport" and "validation" when designing XML representation of bibliographic records (Carvalho and Cordeiro, 2002). The rationale was that the requirements for a XML format that allowed efficient transport of records between systems was incompatible with the complexity of achieving a schema for validation.

A more articulated approach is one that defines a simple schema for transportation, like MARCXML, and defines another schema for representing the rules that define a valid

UNIMARC record. The concept of "valid" here is limited to be what was called a "correct record" (Carvalho et alli, 2004). It corresponds to a record that contains the required set of fields prescribed by the UNIMARC standard to model a given type of bibliographic item and the content of which follows the relevant coding rules and vocabulary types, wherever applicable. Functionally, this is a record that a UNIMARC-aware automated system can extract information from, producing, for instance, indexes or ISBD displays of the contained information. An alternative designation of a "correct" record could be "semantically valid." Limitations of this level of analysis include the possibility that a record can be "correct" in this sense but provide an inappropriate description of the bibliographic item, because it contains semantic errors (misreadings, incorrect recording of dates or names, etc.)

There is a space between the basic structural information that is contained in the MARCXML schema and the high level information required for validation of records by human operators. In that space we find the capacity to automate some basic but relevant tasks: does a record have the mandatory fields?; are the codes introduced in indicators valid?; can I see this record in a more readable form, with field titles and vocabularies expanded?

To automate those tasks we need a machine-readable representation of the rules that make a "correct" UNIMARC record in the sense defined above. In 2004 the National Library of Portugal, in the context of the UNIMARC core activity, contracted BookMARC to develop a prototype of an XML schema that could represent the information contained in the UNIMARC manual. A XML version of the UNIMARC manual should be able to support automated validation of records, translate bibliographic records in human readable terms and providing the manual's contents in different formats. The prototype is available at http://www.bookmarc.pt/unimarc

Description of the Schema

The schema contains a top level element called "FORMAT" that describes the name of the format being represented, the domain and the version. The domain allows for later addition of representation of the information related to non-bibliographic records, like authorities and holdings.

```
<FORMAT name="UNIMARC" domain="Bibliographic" version="1"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="../../schemas/FORMAT.xsd">
<LEADER mandatory="y" repeatable="n">
<RECORD-STATUS name="Record Status" start="6" length="1">
 <OPTION name="Corrected Record" value="c" />
 <OPTION name="Deleted Record" value="d" />
 <OPTION name="New Record" value="n" />
 <OPTION name="Previously Issued Higher Level Record" value="o" />
 <OPTION name="Previously Issued as an Incomplete, Pre-publication Record" value="p" />
 </RECORD-STATUS>

<TYPE-OF-RECORD name="Type of Record" start="7" length="1">
 <OPTION name="Language Materials, Printed" value="a" />
 <OPTION name="Language Materials, Manuscript" value="b" />
 <OPTION name="Music Scores. Printed" value="c" />
 <OPTION name="Music Scores, Manuscript" value="d" />
```

```
  <OPTION name="Cartographic Materials, Printed" value="e" />
  <OPTION name="Cartographic Materials, Manuscript" value="f" />
  <OPTION name="Projected and Video Material (Motion Pictures, Filmstrips, Slides, Transparencies,
Video Recordings)" value="g" />
  <OPTION name="Sound Recordings, Non-musical Performance" value="i" />
  <OPTION name="Sound Recordings, Musical Performance" value="j" />
  <OPTION name="Two-dimensional Graphics (Pictures, Designs etc.)" value="k" />
  <OPTION name="Electronic resources" value="l" />
  <OPTION name="Multimedia" value="m" />
  <OPTION name="Three-dimensional Artefacts and Realia" value="m" />
  </TYPE-OF-RECORD>
…
```

Fragment of the top level of the UNIMARC manual in XML

Inside the "FORMAT" top-level element are the elements corresponding to the record label (or leader) and the fields. For the record label we have a description of each of the positional character and the legal values it can take.

Each field is described through a set of identifier attributes - a field has a *tag* and a *name* - and a set of occurrence attributes - a field may or may not be *mandatory* and/or *repeatable*. *Description* and *examples* elements hold information in (at least almost) human-readable form that can be used in help or documentation systems.

```
<FIELD tag="001" name="RECORD IDENTIFIER" mandatory="y" repeatable="n">
 <DESCRIPTION>This field contains characters uniquely associated with the record, i.e. the control
number for the record of the agency preparing the record.</DESCRIPTION>
<IND1>
 <DESCRIPTION>In conformance with ISO 2709 this field does not have indicators.</DESCRIPTION>
 </IND1>

 <IND2>
 <DESCRIPTION>In conformance with ISO 2709 this field has no indicators.</DESCRIPTION>
 </IND2>

 <SUBFIELD tag="*" name="" mandatory="n" repeatable="n">
 <DESCRIPTION>In conformance with ISO 2709 this field does not contain
subfields.</DESCRIPTION>
 </SUBFIELD>

 <NOTES>There are no restrictions on the form of the record identifier. When the record identifier
consists of or incorporates any form of an International Standard Number, other identifiable number
such a national number, or other data such as country of publication, this information is nevertheless to
be entered in the field specified for that data in addition to recording it as the record identifier in this
field.</NOTES>
 <RELATED tag="*" name="">
 <DESCRIPTION>An agency may be using other separately identified numbers such as ISBN or
National Bibliography Number as a Record Identifier. See Notes on Field Contents above for treatment
in these cases.</DESCRIPTION>
 </RELATED>

<EXAMPLES>
 <EX n="1">
 001 2002-00000000011
 <P />
```

```
A number in the Joint Catalogue file of the South African State Library.
</EX>
…
```

Example of a control field description in the UNIMARC manual in XML

```
<FIELD tag="071" name="PUBLISHER'S NUMBER (SOUND RECORDINGS AND MUSIC)"
mandatory="n" repeatable="y">

 <DESCRIPTION>This field contains publisher's numbers used for sound recordings and publications
of music.</DESCRIPTION>
<IND1 name="Type of Publisher's Number">
 <DESCRIPTION>This indicator specifies the type of publisher's number contained in the field. It may
be used to generate introductory text if a note is to be displayed from this field (EX 2,
4).</DESCRIPTION>
 <OPTION value="0" name="Issue number" />
 <OPTION value="1" name="Matrix number" />
 <OPTION value="2" name="Plate number" />
 <OPTION value="3" name="Other publisher's number (printed music)" />
 <SEEEX N="2" />
 <SEEEX N="4" />
 </IND1>
 <IND2 name="Note Indicator">
 <DESCRIPTION>This indicator specifies whether the agency supplying the record would use the data
to display a note. If a generated note is not required (perhaps because the agency has also entered
details in a 300 or 301 field (EX 3)) then this number should be set to 0.</DESCRIPTION>
 <OPTION value="0" name="Do not make a note" />
 <OPTION value="1" name="Make a note" />
 </IND2>
 <SUBFIELD tag="a" name="Publisher's Number" mandatory="n" repeatable="n">
 <DESCRIPTION>The number as assigned by the agency. Not repeatable.</DESCRIPTION>
 </SUBFIELD>
 <SUBFIELD tag="b" name="Source" mandatory="n" repeatable="n">
 <DESCRIPTION>The agency which assigned the number. Not repeatable.</DESCRIPTION>
 </SUBFIELD>
…
```

Example of a data field description in the UNIMARC manual in XML

Data field elements use the same top level element ("FIELD") but they include other elements like "IND1" and "IND2" for indicator and "SUBFIELD" for subfields. Subfields are described by the same set of identifier and occurrence attributes used in fields description ("tag", "name", "mandatory" and "repeatable").

Fixed length data elements contained in subfields (or fields) are represented through an appropriate element type, "PSUBFIELD" (from "positional subfield"), that adds information about the start and end of the element.

When record elements store only a discreet and finite set of values, an appropriate vocabulary of possible items is composed (provision is made to allow for external vocabularies located by means of an URI). This allows for validation and decoding of bibliographic information for display purposes.

```
<FIELD tag="100" name="GENERAL PROCESSING DATA" mandatory="y" repeatable="n">
<DESCRIPTION>This field contains fixed-length coded data applicable to records of materials in any
media</DESCRIPTION>
<IND1 name="blank (not defined)" />
<IND2 name="blank (not defined)" />
<SUBFIELD tag="a" name="General Processing Data" mandatory="y" repeatable="n">
<DESCRIPTION>All data entered in $a is identified by character position within the subfield.
Conventionally the character positions are numbered from 0 to 35. The character positions defined
must appear in the subfield. Not repeatable.</DESCRIPTION>

<PSUBFIELD n="1" name="Date Entered on File" mandatory="y" start="0" end="7">
<DESCRIPTION>Eight numeric characters in ISO standard form (ISO 8601-1988) for dates:
YYYYMMDD where YYYY represents the year, MM the month with leading 0 if necessary and DD the
day of the month with leading 0 if necessary. The date will usually be the date when the record was
originally created and entered into machine-readable form, to give some idea of the age of the record. A
record corrected because of errors in keying or editing will not have a change of date. On exchange, the
record should also retain its original date.</DESCRIPTION>
<EXAMPLES>
<EX n="1">Character positions 0-7: 19671005 The record was originally entered into a machine-
readable data file on 5th October 1967.</EX>
</EXAMPLES>
</PSUBFIELD>

<PSUBFIELD n="2" name="Type of Publication Date" start="8" mandatory="n" end="8">
<DESCRIPTION>A one-character code indicates the type of date according to eleven categories which
take into account the issuing status of the item being recorded. This code defines the types of dates
which appear in Publication Date 1 (character positions 9-12) and Publication Date 2 (character
position 13-16). These dates are always entered as a four character representation of the year. Note: It
may not be possible always to provide blanks as required for dates from source formats where any digit
has been supplied as an approximation in a corresponding field.</DESCRIPTION>
<VOCABULARY type="local" name="Type of Publication Date">
<ITEM code="a" name="currently published serial">
<DESCRIPTION>Publication date 1 contains the beginning year of publication or coverage if coverage
differs from publication (EX 1). If the beginning date is uncertain, any unknown digit may contain a
blank (EX 2). Publication date 2 contains 9999.</DESCRIPTION>
…
```

Example of the description of positional subfields

The schema goes further than formalizing the structural rules that allows automated validation of records. It also has elements for encapsulating the abundant examples, and cross-references that are found in the UNIMARC manual. These elements ("EXAMPLES", "SEEEX" and "RELATED") do not play a role in validation but allow for rendering the manual is useful forms, like formatted HTML or PDF, with links between the various examples, related fields and descriptions.

```
<FIELD tag="208" name="MATERIAL SPECIFIC AREA: PRINTED MUSIC SPECIFIC
STATEMENT" mandatory="n" repeatable="n">
<DESCRIPTION>This field contains a statement, transcribed from the item, relating to the particular
format of a printed or manuscript music item. The field corresponds to the ISBD(PM) Printed Music
Specific Area.</DESCRIPTION>
<IND1 name="blank (not defined)" />
<IND2 name="blank (not defined)" />
<SUBFIELD tag="a" name="Printed Music Specific Statement" mandatory="n" repeatable="n">
<DESCRIPTION>The statement related to the particular format of the item in hand, taken from the
```

```
title page (EX 1, 2). Not repeatable.</DESCRIPTION>
  <SEEEX N="1" />
  <SEEEX N="2" />
  </SUBFIELD>
 <SUBFIELD tag="d" name="Parallel Printed Music Specific Statement(s)" mandatory="n"
repeatable="y">
  <DESCRIPTION>Any additional statements in a different language and/or script, taken from the title
page (EX 2). Repeatable for each additional statement.</DESCRIPTION>
  </SUBFIELD>
 <NOTES>
  Parallel data: Parallel data appearing on the item, i.e. data repeated in another language or script, is
denoted in ISBD by the use of the equals sign followed by a space = at the start of any block of parallel
data. In UNIMARC '= ' is included in the data whenever required by ISBD, preceding and in the same
subfield as the data to which it relates (EX 2).
  <SEEEX N="2" />
  </NOTES>
 <RELATED tag="125" name="CODED DATA: SOUND RECORDINGS AND MUSICAL SCORES">
  <DESCRIPTION />
  </RELATED>
 <EXAMPLES>
  <EX n="1">208 ##$aMiniature score</EX>
  <EX n="2">208 ##$aOrchester-Partitur$d= Full score</EX>
  </EXAMPLES>
  </FIELD>
```

Example of context information: notes, examples, cross-references

One of the complex aspects of coding in XML is dealing with conditional rules. For instance, cartographic materials should have a 206 field with a statement of scale. In UNIMARC this means that if character 7 of the Record Label ("Type of Record") is "e" ("Cartographic Materials, Printed") of "f" ("Cartographic Materials, Manuscript"), then field 206 is mandatory, otherwise it is not. This was formalized by a special element, called "MANDATORY-CONDITION".

```
<FIELD tag="206" name="MATERIAL SPECIFIC AREA: CARTOGRAPHIC MATERIALS -
MATHEMATICAL DATA" mandatory="c" repeatable="y">
  <DESCRIPTION>The statements of scale, projections, coordinates and equinox of a cartographic
item.</DESCRIPTION>
  <IND1 name="blank (not defined)" />
  <IND2 name="blank (not defined)" />

- <MANDATORY-CONDITION>
  <RULE>substring($leader, 7, 1) = 'e' or substring($leader, 7, 1) = 'f'</RULE>
  <NOTES>This field is mandatory for cartographic material.</NOTES>
  </MANDATORY-CONDITION>
…
```

Example of a conditional rule for validation

The UNIMARC-XML manual in use

The specific structure of the UNIMARC manual in XML was designed so that it could be used efficiently with standard XML transformations. XSLT transformations are a standard way to transform XML documents into other XML documents. Documents that specify

XSLT transformations are called XSLT stylesheets. The prototype includes stylesheets that allow for HTML formatting of the manual, validation of records and HTML formatting of records.

The *FORMATtoHTML.xsl* stylesheet generates an HTML version of the UNIMARC manual. Variations of this stylesheet can be produced to obtain different styles of output. A similar stylesheet could be designed to produce XML specifications for a PDF output.

RECORD LABEL	
Mandatory	Yes
Repeatable	No

Record Status: From 6 Length 1	
Vocabulary	
c	Corrected Record
d	Deleted Record
n	New Record
o	Previously Issued Higher Level Record
p	Previously Issued as an Incomplete, Pre-publication Record

Type of Record: From 7 Length 1	
Vocabulary	
a	Language Materials, Printed
b	Language Materials, Manuscript
c	Music Scores. Printed
d	Music Scores, Manuscript
e	Cartographic Materials, Printed
f	Cartographic Materials, Manuscript
g	Projected and Video Material (Motion Pictures, Filmstrips, Slides, Transparencies, Video Recordings)
i	Sound Recordings, Non-musical Performance
j	Sound Recordings, Musical Performance

Example of an HTML rendering of the UNIMARC manual in XML

700: PERSONAL NAME - PRIMARY INTELLECTUAL RESPONSIBILITY		
Mandatory	No	
Repeatable	No	
First Indicator	blank (not defined)	
Second Indicator	Form of Name Indicator **See examples:** *5*	
	Vocabulary	
	0	Name entered under forename or direct order
	1	Name entered under surname (family name, patronymic, etc.) *This indicator specifies whether the name is entered under the first occurring name (forename) or in direct order or whether it is entered under a surname, family name, patronymic or equivalent, usually with inversion (EX 5).*
Description	This field contains the name of the person considered to have primary intellectual responsibility for a work in an access point form, provided that the record is created according to cataloguing rules recognising the concept of main entry. Not repeatable. It may not occur in the same record as field 710 CORPORATE BODY NAME PRIMARY INTELLECTUAL RESPONSIBILITY or field 720 FAMILY - NAME PRIMARY INTELLECTUAL RESPONSIBILITY, since a record can have only one access point with primary intellectual responsibility. If the concept of main entry does not exist in the cataloguing rules or the source format does not distinguish it, this field does not occur: field 701 is used for all personal headings (see Related Fields).	

Example of an HTML rendering of the UNIMARC manual in XML

The power of XML stylesheets really shows when we use the XML UNIMARC manual to validate or make readable rendering of bibliographic records ("Explaining records"). These operations correspond to a merge between an XML representation of the bibliographic record and the XML representation of the manual. This can be done through stylesheets in a two-step process. To illustrate the possibilities created by the UNIMARC manual in XML we emulate the function of the stylesheets provided with the MARCXML toolkit for validation and HTML formatting (see Corey, 2004).

In the supplied examples we assume UNIMARC records represented in XML using the MARCXML slim schema. The prototype supplies two stylesheets for this purpose.

The "HTMLFormaterGenerator.xslt" is a stylesheet that, when applied to the UNIMARC manual in XML, generates another stylesheet that is able to format MARCXML bibliographic records in HTML. The generated stylesheet, when applied to a bibliographic record produces HTML output.. The stylesheet generated by HTMLFormaterGenerator is functionally equivalent to the similar stylesheet included in the MARCXML toolkit ((http://www.loc.gov/standards/xmlmarc).

LEADER
    Record Status: c
    Type of Record: a
    Bibliographic Level: m
    Hierarchical Level Code:
    Encoding Level: 1
    Descriptive Cataloguing Form:
RECORD IDENTIFIER

450981
VERSION IDENTIFIER
            20030517004400.0
            …..
COUNTRY OF PUBLICATION OR PRODUCTION
            Country of publication: PT
TITLE AND STATEMENT OF RESPONSIBILITY
            Title Proper: UNIMARC
            Other Title Information: Manual de operações
            First Statement of Responsibility: Fernanda Maria Guedes de Campos, José Carlos Sottomayor
PUBLICATION, DISTRIBUTION, ETC
            Place of Publication, Distribution, etc: Lisboa
            Name of Publisher, Distributor, etc.: Biblioteca Nacional,
            Date of Publication, Distribution, etc.: 1986
PHYSICAL DESCRIPTION
            Specific Material Designation and Extent of Ite: 1 pasta (pag. var.)
            Dimensions: 32 cm
PERSONAL NAME - PRIMARY INTELLECTUAL RESPONSIBILITY
            Entry Element: Campos,
            Part of Name Other than Entry Element: Fernanda Maria Guedes de,
            Dates: 1949-
PERSONAL NAME - ALTERNATIVE INTELLECTUAL RESPONSIBILITY
            Entry Element: Sotomaior,
            Part of Name Other than Entry Element: José Carlos,
            Dates: 1953-
            Relator Code: 070

Formatted Record with stylesheet derived from the UNIMARC XML manual

It is important to note that no special software is used to produce the final result. All is needed is a standard XML package capable of applying XSLT transformations. The process is completely data driven in the sense that from the UNIMARC manual in XML it is possible to derive, by standard XSLT processing, a stylesheet that is able to format records in a human readable way. This also means that alternative XML formats for records could also be handled, by coding the appropriate stylesheets. The stylesheet is also language neutral, because it just uses the description of items as provided by the UNIMARC manual in XML. A UNIMARC manual in a different language, as long as it used the same schema, would produce a record "explained" in that language.

A similar procedure is used to produce validating stylesheets. A validating stylesheet is a stylesheet that when applied to a bibliographic record produces a XML document that describes existing errors, if any. The "RecordValidatorGenerator.xslt" stylesheet is applied to the UNIMARC manual in XML and generates another stylesheet that, when applied to a record, validates that record in terms of mandatory fields and sublfields and of vocabularies. The MARCXML toolkit of the LoC includes such a stylesheet. An example of the output of this stylesheet is the following:

```
<validationReport>
 <warning type="UnknownTag">
 <datafield tag="095" ind1="" ind2="">
 <subfield code="a">PTBN00653291</subfield>
 </datafield>
```

```
  </warning>

<warning type="UnknownTag">
<datafield tag="801" ind1="" ind2="0">
 <subfield code="a">PT</subfield>
 <subfield code="b">BN</subfield>
 <subfield code="g">RPC</subfield>
 </datafield>

 </warning>

<warning type="UnknownTag">
<datafield tag="998" ind1="" ind2="">
 <subfield code="a">MPA11-1</subfield>
 </datafield>

 </warning>

<warning type="UnknownTag">
<datafield tag="966" ind1="" ind2="">
 <subfield code="l">MPAA</subfield>
 <subfield code="m">MPAA</subfield>
 <subfield code="s">1622/M</subfield>
 </datafield>

 </warning>

 </validationReport>
```
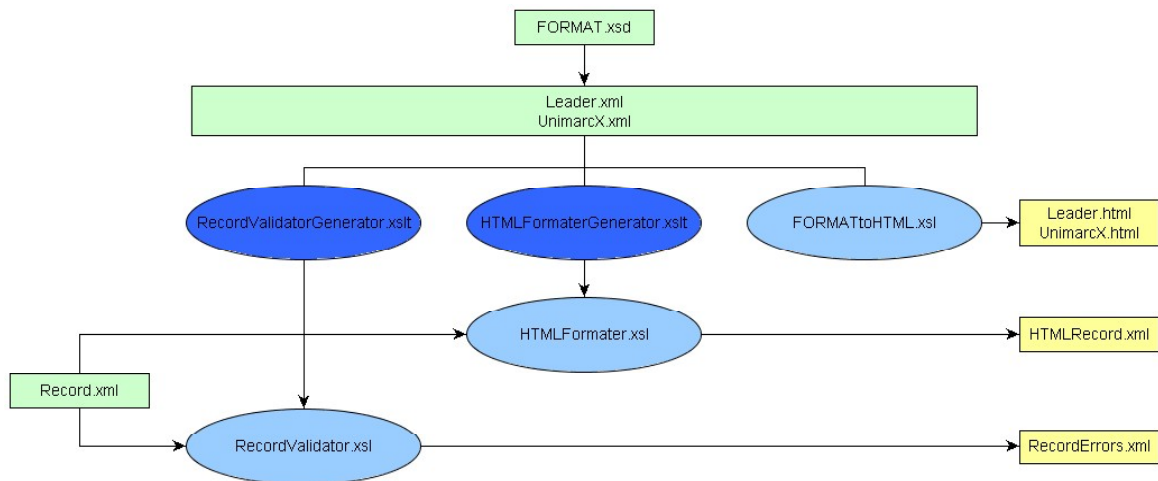
Example of validatin of records from information in the UNIMARC manual in XML

The following diagram exemplifies the flow of transformations in the UNIMARC XML prototype package.

Further possibilities

The UNIMARC manual in XML prototype should be considered as a proof of concept. It demonstrates that using XML to represent semantic rules of UNIMARC is possible and that sophisticated processing is possible through standard means like XSLT transformations.

Further possibilities of usage would include the automatic updating of software systems by importing a new version of the UNIMARC manual. Another area that would be simplified is localization of systems, in different languages, as long as local versions of the UNIMARC manual in XML were provided. Software processing UNIMARC records would also be able to perform rather complex validation without much effort.

It is taken for granted that the major advantage of having and XML representation of bibliographic records in the possibilities created by processing the records using the sophisticated capabilities of XSLT. In our view the representation on the rules ("the manual") in XML brings further possibilities, precisely because XSLT stylesheets can be produced automatically for some purposes, like validation and formatting, with more ease and less maintenance problems.

Open Issues

When we speak of the UNIMARC manual in XML prototype we refer to the schema that was developed. The transformations that are provided with the package are there just to exemplify usages of the schema and were designed to show that relevant functionality could be in part automatically produced from a XML representation of the rules underlying the format.

Open issues relate to the schema and not the examples given. The major issue regards conditional rules, like the one exemplified in the text. It is not clear at this point if all possible rules of that type are readily handled by the current schema.

Another issue regards the "neutrality" of the manual's schema regarding the XML format for bibliographic records. In our examples we assumed records in MARCXML/slim format. But there is not, at this moment, a standard for the representation of UNIMARC in XML, and different proposals are on the table. In principle the schema for the manual could be made independent of the schema of records. This is what we tried here, encapsulating in the stylesheets the way the rules apply to records. But validation information would be easier to encapsulate in the manual if a known XML format for the records was assumed.

Another final aspect relates to the other, non-bibliographic, formats under the UNIMARC umbrella. It would be necessary to test if the schema can also handle those other types of records and their associated rules, or if further elaboration is necessary.

A less technical but relevant aspect of providing the UNIMARC manual in XML relates to issues of licensing and copyrighting, that would have to be decided and formalized.

Acknowledgments

References

Carvalho, J; Cordeiro, MI (2002), "XML and bibliographic data: the TVS (Transport, Validation and Service) model". http://www.ifla.org/IV/ifla68/papers/075-095e.pdf

Carvalho, J; Cordeiro, MI; Vieira, M; Lopes, A. (2004), "Meta-information about MARC: an XML framework for validation, explanation and help systems", ***Library Hi-tech***, **vol. 22, n.2, 2004**.

Keith, Corey (2004), "Using XSLT to manipulate MARC metadata", ***Library Hi-tech***, **vol. 22, n.2, 2004.**

More information

http://www.bookmarc.pt/unimarc