

## How to express MARC in XML : ELAG 2004. Workshop 10 Report

Participants:

- Liv Aasa Holm, JBI-HIO, Norway;
- Christer Larsson, The Royal Library, LIBRIS Department, Sweden;
- Dan Matej, CIMEC - Institute for Cultural Memory, Romania;
- Anne Munkebyaune, BIBSYS, Norway;
- Mona-Lise Pedersen, BIBSYS, Norway;
- Nils Pharo, Oslo University College, Norway.

### A. Why XML ?

First issue discussed was: (for the bibliographic systems) is XML (really) useful or it is (just) fashionable ?

After a succinct examination, we concluded that XML could be a useful metalanguage for the manipulation/storing of catalogue data, and the main merits of it we detected were:

- it has a flexible syntax, i.e. it has much "expressive power";
- it allows useful syntactic constraints;
- it allows (unrestricted) hierarchies in a record;
- there is a lot of tools available;
- it is here to stay (?).

### B. Problems with the MARCs ?

The second issue debated was: do we have problems with (the many varieties of) the MARC format(s) ?

The (main) problems we detected were:

- the "1 to 1 principle" is not observed (i.e. the records are not "normalized", that is, a record contains data about several entities, e.g. the book and its authors); the "bright side": the MARC record is "self-contained" !
- it is not too flexible, i.e. it is almost flat, i.e. it allows only 2 (or 3 ?) hierarchical levels, that is, it does not allow a good control of the granularity of data; [on the other hand — Ole Husby contributed — it is too flexible, i.e. (as we understand it) it allows (too often) alternative solutions to a problem];
- some tags (e.g. those for the headings) express two different things: a) the nature of the related entity and b) the kind of relationship of that entity with the "host" record; this leads to a bigger MARC "schema" than necessary;
- it does not allow (naturally) multilingual data within a record, i.e. for the fields with values in the language of the cataloguing.

### C. Aim: to devise an XML-based catalographic format

As a consequence of the previous two conclusions, the idea of an XML-based bibliographic format was explored.

One approach, i.e. to "mechanically" express MARC in XML was already done by the Library of Congress: *marcxml*, see [www.loc.gov/standards/marcxml/](http://www.loc.gov/standards/marcxml/). We agreed it is not a very interesting solution. It is just a change of syntax.

We considered then a more promising approach: the integration of authority, bibliographic and holdings MARCs in a new format (i.e. a BML [bibliographic MARC-up language] ?) [we codename-it MARCX].

We discussed two major use-cases of such a language:

- A. as an internal (database) format;
- B. as a transportation/serialization format.

Two main "transport scenarios" were identified:

- to/from union catalogues (or inter-catalogue communication): "normalized" files, i.e. records of instances of "base" entities, plus records for their relationships, that is, bibliographic records, authority records and item/holdings records, along with distinct relationship records (i.e. in the manner of a topic map); a variation could be: records containing "FRBR families" of bibliographic objects, e.g. works with their expressions;
- to client applications, for presentation (i.e. display): un-normalized, self-contained (MARC-like) records.

## **E. The "integrating" framework: FRBR**

We agreed that FRBR could be a good integrating framework for MARCX, that is, its schema (and/or DTD) should include elements/types for:

- works;
- expressions;
- manifestations;
- items;
- persons;
- corporate bodies;
- concepts;
- objects;
- events;
- places;

as well as for:

- subject headings;
- relationships.

And — maybe — even more.

## **F. Relationships: identifiers**

In order to record properties of the entities within the bibliographic universe, we need to relate the instances of the entities and to relate the elements of an instance. Hence

the need for identifiers. We discussed:

- the need for unique identifiers within a file;
- the need for global unique identifiers;
- the need for large amounts of unique identifiers, i.e. the need for automatic identifier generation.

Two options: URIs and GUIDs [Global Unique Identifiers] were considered.

## **G. Relationships: options**

Two ways of expressing relationships between the entity instances were considered.

a) Reified relations (Topic Maps like), i.e. explicit relationships, external to both related instances. They may look like this:

```
<relation type="some type">
  <source>source-id</source>
  <target>target-id</target>
  <otherRelationProperties>
    ...
  </otherRelationProperties>
</relation>
```

b) Pointers to the targets (within source). They may look like this:

```
<entity ....>
  ...
  <relation type="some type">
    <target>target-id</target>
    <otherRelationProperties>
      ...
    </otherRelationProperties>
  </relation>
  ...
</entity>
```

## **H. Relationships: the "type problem"**

We discussed then the best way to record the type of a relation. We considered:

a) the type expressed as attribute; e.g.

```
<relation type="author">person id</relation>
```

b) the type as element; e.g.

```
<relation>
  <type>author</type>
  <target>person id</target>
</relation>
```

The second option seems to be more convenient for "ontology controlled" types.

## I. Types/elements: inner structure

Discussing the issue of the bibliographic records proper, i.e. manifestations (editions, etc.), the group considered that that it is not convenient to preserve the traditional MARC blocks. Moreover, we agreed that it would be more useful to "re-group" the data elements by their nature, e.g. 'title' and 'notes on title'. Also we think that it is advisable to use as many hierarchical levels as necessary (but not more :-).

## J. MARCX files: the general pattern

We speculated a bit about what the files could look like. As the first approximation, they could look like this:

```
<file ...>
  ...
  <entity ....>
    <identifiers>
      ...
    </identifiers>
    <description>
      ...
    </description>
  </entity>
  ...
  <relation ...>
    ...
  </relation>
  ...
</file>
```

## K. Language independence

One of the important problems this format could solve is that of "language independence" for multilingual records (which should be the norm in the international union catalogues). By "language independence" we mean the property of allowing the automatic detection of the language of the literals. The XML syntax allows easily the exposure of the language of each piece of text.

So, we devised the low-level element "localized text (*ltext*), with (at least) the attributes:

- language;
- script;
- transliteration standard.

E.g.

```
<someElement ...>
  <ltext lang="en" script="latin">
    What the hell is going on ?
  </ltext>
  <ltext lang="fr" script="latin">
    Mais qu'est qui se passe ?
  </ltext>
```

```

    <lttext lang="ro" script="latin">
      Ce dracu se întâmplă ?
    </lttext>
  </someElement>

```

Also, we can conveniently expose the language of the material, in areas where the cataloguing rules ask for that language, such as the 'title and statement of responsibility'. E.g.

```

<titleAndResponsibility lang="en">
  <title type="proper">
    Romeo and Juliet
  </title>
  <title type="parallel" lang="fr">
    Romeo et Juliette
  </title>
  <title type="translated" lang="de">
    Romeo und Julietta
  </title>
  <statementOfResponsibility>
    Shakespeare
  </statementOfResponsibility>
</titleAndResponsibility>

```

Or, what in MARC 21 is expressed rather acrobatically as:

```

...
245 10$6880-01/(N$aРомео и Джульетта$cСергей Сергеевич Прокофьев
...
880 10$6245-01/(B$aRomeo i Djulietta$cSerghey Sergheevici Prokofiev
...

```

could be expressed more naturally as:

```

<titleAndResponsibility lang="ru" script="cyrillic">
  <title type="proper">
    <lttext>
      Ромео и Джульетта
    </lttext>
    <lttext script="latin">
      Romeo i Djulietta
    </lttext>
  </title>
  <statementOfResponsibility>
    <lttext>
      Сергей Сергеевич Прокофьев
    </lttext>
    <lttext script="latin">
      Serghey Sergheevici Prokofiev
    </lttext>
  </statementOfResponsibility>
</titleAndResponsibility>

```

[provided this is the right transliteration :-)]

## **L. Final remarks**

Our answer to the "classical" ELAG question "To tag or not to tag ?" is: to tag !

To tag in MARCX could offer:

- finer (and more controllable) granularity;
- less redundancy;
- more compact records;
- more human-readable records
- language-independent (i.e. multilingual) records.

Also it could:

- reduce the number of duplicate records (generated during exchange), due to the strict observation of "1 to 1 principle", i.e. not including headings in the bibliographic records, linking to authority records instead;
- allow the use of a lot of ready-made tools.

Open question left: (unless the MARCs) could MARCX become a lingua franca ?

[Trondheim, June 9-10, 2004]