Submitted on: 01.07.2015

Our Standards vs Their Standards: Development and Re-Use of Non-Library Standards in the Cultural Heritage Domain

Lars G. Svensson
Information Infrastructure and Preservation, Deutsche Nationalbibliothek, Frankfurt am Main, Germany.
E-mail address: l.svensson@dnb.de

Abstract:

When exchanging data between systems, it is important to use well-defined standards that are widely acknowledged and have a large user base in order to make the data interoperable.
This paper looks at how three kinds of data are transported in MARC 21 and UNIMARC: floating point numbers, (birth) dates and geographic coordinates. The encoding of this data is compared to existing industry standards and finally discussed in the context of how this information should be recorded using the upcoming library standard RDA and the role of library systems as mediators between cataloguing code, local conventions and exchange formats.

Keywords: Standards

1 INTRODUCTION

Libraries have a long tradition of exchanging information about the resources they curate. A prerequisite for this kind of information exchange to be effective is the use of a common set of rules, not only for the resource description, but also for how this description is packaged and transmitted. If this was fairly straightforward 50 years ago – a typed description on a catalogue card being sent by surface mail – we now have electronic descriptions packaged as MARC records sent over the internet using HTTP or FTP. This development would not have been possible without a shared set of rules, what we commonly call standards. So far, many of the standards being used for bibliographic exchange have been specific to the library community, e. g. the development of cataloguing rules, the ISBD format or the creation of the family of MARC formats based on ISO 2709. Currently, there are two lines of development forcing us to look closer at what standards we use:

1. Bibliographic information is today not only intended for human consumption but must also be ready for machine-interpretation
2. There is an increasing interest in use and re-use of data from libraries – both authority data and title data – by organisations outside of the traditional library domain

The first development means that we need to store our information in a way that machines can read and act on, in a sense we're turning bibliographic information into data, which among other things will enable us to build better user interfaces. The second line of development makes it necessary to discuss what standards we must use in order to enable non-library systems to make sense of our data. Do we use our standards or their standards?
The rest of this paper is structured as follows: Section 2 analyses three types of data where current library standards for data creation and exchange make interoperability with outside systems more difficult than necessary: dates, floating point numbers and geographic coordinates. Section 3 looks at the instructions in upcoming content standard RDA and how they relate to the exchange formats. Section 4 finally is a call-for-action: what do librarians need to do in order to make their data more interoperable.

2 CASE STUDIES

This chapter looks at three cases where current library standards for data creation and exchange make interoperability with outside systems more difficult than necessary: dates, floating point numbers and geographic coordinates. All three cases have in common that they deal with information that is obviously suitable for machine consumption (and possibly also created by machines), as opposed to text such as title or publication statements that are transcribed verbatim from the resource at hand.

2.1 Floating point numbers

One commonly performed task when processing data is comparing numbers, e. g. for creating a sorted list or for deciding which rule applies for the processing of a specific object. The most commonly used norm for floating point numbers (FPN) is IEEE 754[1] (also published as ISO/IEC/IEEE 60559:2011) where a FPN is defined as $(-1)^s \times c \times b^q$ where b is the base (2 or 10), s = a sign (0 or 1), c = a significand (or "coefficient") and q = an exponent. An example is -12345×10^{-3} or 98.7654×10^{6}, compactly written as -12345e-3 resp. 98.7654e6. It is important to note that the standard mandates that the significand (or mantissa) uses a radix *point*, i. e. the correct format for a number with a fraction is "123.456" (which is the common formatting e. g. South-East Asia and in the English speaking part of the world), whereas continental Europe, large parts of Africa and northern Asia use a radix comma, i. e. "123,456" (both cases indicating a number x: 123 < x < 124). In scientific data processing, the use of a radix point is prevalent, being the variant used by most computer systems and for the internal representation in most programming languages.[2]

On recent case where floating point numbers have entered into library exchange formats is in field 883 in MARC 21: "Machine-generated Metadata Provenance", introduced in September 2012.[3] The contents of this repeatable field are really meta-meta-data, giving information about how contents of other fields were created. Given that a growing number of libraries are experimenting with the assignment of subject headings derived from full text

[1] https://en.wikipedia.org/wiki/IEEE_754

[2] Most programming languages offer the possibility to localise the way numbers are formatted in the user interface. This must not be confused with how the numbers are represented internally.

[3] http://www.loc.gov/marc/bibliographic/bd883.html

analysis it is growing more and more important for consumers to know if a certain subject heading was assigned intellectually or by a machine (and when assigned by machine: using which algorithm and with which confidence).

The confidence with which the machine generated a certain piece of information is transported in 883 $c: “Confidence value”. The description tells us that this subfield “contains a floating point value between 0 and 1,” and goes on to specify that “either a comma or a point may be used as a decimal marker.” This of course has the advantage that a human entering a value into 883 $c could do that in the format they are most accustomed to. Given, however, that we can assume that the contents are almost always created and interpreted by machines, we would expect the use of “floating point number” in the sense of IEEE 754 which mandates decimal *point*. If you try to create a FPN in Java, e. g. “0,456e-3”, by using `float f = Float.valueOf(“0,456e-3”)`, it will not accept the input but throw a `NumberFormatException`.[4] Since the subfield contents can only be a number between 0 and 1, we can of course inspect the string to figure out if the data is formatted using a decimal point or a decimal comma and then convert it correspondingly. This, however, requires extra processing on the data consumer side and certainly is not in the sense of Postel’s law: “Be rigid in what you send and liberal in what you accept”.

If you implement 883 in your MARC 21 data exchange, please use the decimal point as a fraction mark. It will make life easier for your customers wanting to select only those automatically created subject headings that were created with a confidence larger than e. g. 0.8.

2.2 (Birth) Dates

Dates are prevalent in all kinds of bibliographic and authority information, be it a publication date, the birthdate of a person or the date on which some machine-processing of data in an ILS was performed. This makes it even more astonishing that the recording, storing and exchanging information about time-bound events is regulated very loosely both in cataloguing instructions and in exchange format documentation. The examples here focus on birth dates of persons.

A common example of how dates are used in end user applications is to offer the possibility to search for events within a certain timeframe (e. g. creating a list of all people born between 1850 and 1899) or to show events on a timeline. Most major programming languages offer at least basic support for comparing and sorting dates, in order to be able to process the data, they need to be able to parse the string representing the date in order to create an internal representation. The most widely adopted format for dates is ISO 8601 “Data elements and interchange formats – Information interchange – Representation of dates and times,”[5] where dates (and time points) are represented in the order year, month, day and the parts can be separated by dashes (2015-08-19, extended format) or written together (20150819, basic format). There is a specific profile of ISO 8601 developed by the W3C that aims “to simplify the use of ISO 8601 in World Wide Web-related standards.”[6] Of particular interest to the cultural heritage community is the Extended Date/Time Format (EDTF) 1.0,[7] a date format proposed by the Library of Congress that on the one hand acts as a profile of ISO 8601 by narrowing down the number of supported date/time formats, and on the other hand also extends that specification by offering syntax for uncertain and approximate dates (e. g.

[4] For a thorough documentation of how to represent FPNs in Java, cf. https://docs.oracle.com/javase/7/docs/api/java/lang/Double.html#valueOf%28java.lang.String%29

[5] https://en.wikipedia.org/wiki/ISO_8601

[6] http://www.w3.org/TR/NOTE-datetime

[7] http://www.loc.gov/standards/datetime/pre-submission.html; all examples are taken from this source.

"1984~" for "'approximately' the year 1984"), unspecified dates ("199u" for "some unspecified year in the 1990s."), sets of dates ("[1667,1668,1670..1672]" meaning "one of the years 1667, 1668, 1670, 1671, 1672" and "{1667,1668,1670..1672}" meaning "all of the years 1667, 1668, 1670, 1671, 1672" (i. e. not the interval 1667-1672, but an event taking place each of the years). The proposal is very promising, what is lacking is a section on the applicability of the standard datatypes from XML schema[8] with the extended dates: Would e. g. "1984~" still have the datatype xsd:date or would it be necessary to introduce new types for the extensions?

2.2.1 MARC 21 Authority

In the MARC 21 Authority format, there are at least three places where the birth date of person can be recorded: 046, 100 and 548.[9] At all three places, different date formats are used.

The most commonly used field for birth (and death) dates in MARC 21 is probably the field 100: "Heading-Personal Name".[10] The date is entered in 100 $d and the format is fairly liberal: "Dates of birth, death, or flourishing or any other date used with a name. A qualifier used with the date (e.g., b., d., ca., fl., ?, cent.) is also contained in subfield $d." The examples include "1678-1763", "d. 45 B.C" and "1240 or 41-ca. 1316". In the case of 100 $d, the liberal date format can be considered a feature rather than a bug, since 100 is intended for human consumption rather than for machine processing, so any date format that is understandable for a patron browsing a set of person authorities is acceptable. We must not, however, expect that machines can make sense of the information contained.

The second possibility in MARC 21 is 548: "See Also From Tracing-Chronological Term". The chronological term goes into 548 $a for the specification is very brief: "Chronological term used as an entry element," which essentially allows any format with the addition that it applies to information "occurring in chronological term headings constructed according to generally accepted thesaurus-building conventions (e.g., *Faceted Application of Subject Terminology* (FAST))," and that "chronological terms include dates that are used as the lead element of subject or added entry access fields in bibliographic records." The use of 548 for transporting birth and death dates is fairly non-standard; it seems that the only institution using this possibility is the Deutsche Nationalbibliothek, the reason being that it is the field that supports the largest range of relation types between the described entity and the chronological term.

A further place to transport dates in MARC 21 Authority is 046: "Special Coded Dates", last updated in April 2015.[11] This field provides subfields for e. g. birth and death dates, establishment and termination dates or start and end of a period. For data exchange purposes, the most important point is that in all subfields,

> date and time are recorded according to *Representations of Dates and Times* (ISO 8601) in the pattern yyyy, yyyy-mm, or yyyymmdd (4 for the year, 2 for the month, and 2 for the day) unless subfield $2 (Source of date) specifies another date scheme.

This means that a consuming application can first look at 046 $2 to find out how the dates are encoded and then interpret the dates using standard date parsers, defaulting to an

[8] http://www.w3.org/TR/xmlschema11-2/#built-in-primitive-datatypes

[9] A full list of where dates are used in MARC 21 is available at http://www.loc.gov/marc/yr2000.html, for authority data those fields marked with "AD".

[10] http://www.loc.gov/marc/authority/ad100.html

[11] http://www.loc.gov/marc/authority/ad046.html

ISO 8601 parser if $2 contains no data. This makes it extremely easy to build new functionality on top of the library information.

2.2.2 UNIMARC Authorities

In the UNIMARC Authorities format, there are two places to put birth dates: 200 $f: “Authorized Access Point – Personal Name – Dates” and 640 $f/$i: “Place(s) and Date(s) Associated with the Entity -- Date of beginning or unique date”.

The guidelines for 200 $f are similar to those for 100 $d in MARC 21 Authorities: the field is not structured and is more geared towards human inspection than machine consumption (UNIMARC 2009, 111-113). This is in sharp contrast to the well-defined semantics of 640 (UNIMARC 2009, 229-232). For both subfields describing dates in 640 ($f and $i) UNIMARC mandates a field length of exactly ten positions. The first one (position 0) contains era information where a “#” indicates CE (Common Era) and “-“ BC (Before Common Era). The positions 1-8 contain

> eight numeric characters in ISO standard form (ISO 8601) for dates YYYYMMDD where YYYY represents the year, MM the month with leading 0 if necessary and DD the day of the month wich leading 0 if necessary. If any digit in a year, month and/or day is unknown, the character position should be blank. [X 230]

The last character (position 9) transports reliability information, “#” indicating that the information is certain and “?” that it is an uncertain date.

Again, the explicit reference to ISO 8601 as the date format makes it easy for consuming applications to interpret the data correctly.

2.3 Geographic Coordinates

When cataloguing maps and other resources describing spatial resources, it is common to also record the geographic coordinates describing the spatial extent of the item at hand. Likewise we increasingly find geospatial information in authority data describing geographic entities such as countries, cities or mountains. Through new services such as Open Streetmap, there is much interest in geolocated information, since it allows the data to be visualised in new user-friendly ways. This is also true for bibliographic data, where the combination of bibliographic resources linked to authority data containing geographic coordinates makes it possible to e. g. find documents related to a certain geographic region, or to plot the distribution area of a regional newspaper on a map as in the upcoming newspaper portal for the German Union Catalogue of Serials (Zeitschriftendatenbank, ZDB).[12] Geographic coordinates are more complex than they seem at first glance: what is generally perceived as a question of latitude and longitude, is to experts a matter of different ellipsoids and coordinate reference systems (CRS), where a CRS is a combination of a specific ellipsoid, the axis order (lat/long vs. long/lat) and other information. This complexity is generally not mirrored in the bibliographic information, e. g the content standards do not explicitly mention the ellipsoid used, which means that most people will assume WGS 84 since that is the most commonly used one.

[12] http://www.dnb.de/EN/Wir/Projekte/Laufend/zdbWeiterentwicklung.html

2.3.1 MARC 21 Authority

In MARC 21 Authority, geographic coordinates are exchanged in field 034: Coded Cartographic Mathematical Data.[13] In the simplest case, the coordinates describe a bounding box giving the western- and easternmost longitudes and the northern- and southernmost latitudes (subfields $d, $e, $f and $g). The format of the coordinates is not very rigid, but only specifies that "the coordinates may be recorded in the form *hdddmmss* (hemisphere-degrees-minutes-seconds), however, other forms are also allowed, such as decimal degrees." It is also possible to specify more complex geometries such as polygons using inner and outer G-rings. In those cases, the rings are described by entering latitudes and longitudes pairwise into subfields $s and $t; a second indicator of "0" denotes an outer, "1" an inner ring, there is no specification if the coordinates should run clockwise or counter clockwise. The description of e. g multipolygons is not specified.

2.3.2 UNIMARC Authorities

UNIMARC records geographic coordinates in field 123: "Coded Data Field: Territorial or Geographical Name" (UNIMARC 2009, 77f.). UNIMARC can only describe bounding boxes by recording western- and easternmost longitudes and northern- and southernmost latitudes in $d, $e, §f and $g. All those subfields are fixed-length, with position 0 being the hemisphere, 1-3 the degrees, 4-5 the minutes and 6-7 the seconds. There is no support for more complex geometries.

2.3.3 Wide-spread non-library standards

The most important standards body for geospatial information is the Open Geospatial Consortium (OGC). Among other specifications, the OGC has published two text formats for describing geometries, "Geography Markup Language" (GML), an XML grammar, and "Well-Known Text (WKT)" a string markup format. Both those formats are well-defined, both in terms of how to describe complex geometries such as multipolygons and of how to specify which CRS the data uses, including a default to CRS 84 if no CRS is given. WKT has a large implementation base and is supported by most large database systems and is also supported by the GeoSPARQL standard thus making WKT data easy to integrate with different search scenarios including sharing on the (sematic) web.

3 DATES AND GEOGRAPHIC COORDINATES IN RDA

The upcoming cataloguing code RDA (Resource Description and Access) goes a long way to specify how to describe library resources and authorities. In this section we have a brief look at the instructions RDA provides on how to enter dates and geographic coordinates (there is currently no section in RDA where the entry of floating point numbers is described).

3.1 Birth dates

The relevant section in RDA advices the cataloguer to "record dates in terms of the calendar preferred by the agency creating the data" and to "record a date associated with a person by giving the year." An option used by PCC, BL and D-A-CH is to "add the month or month and day in the form *[year] [month] [day]* or *[year] [month]*. Record the month in a language and script preferred by the agency creating the data." (RDA 2015, §9.3.1.3)

[13] http://www.loc.gov/marc/authority/ad034.html

Whereas those rules make it very easy for a cataloguer to enter the information and for a (local) library patron to understand it, this instruction introduces a tension between the cataloguing code and the exchange format. If the date entered by the cataloguer is stored verbatim in the exchange format, it will be difficult for a machine to interpret the data correctly, particularly if the data is re-used by a consumer not knowledgeable of the context in which the data was created. The first is the calendar referred to as “the calendar preferred by the agency creating the data”, the second is the recording of the month “in a language and script preferred by the agency creating the data”. As long as the instruction does not tell the cataloguer to also record which calendar is meant (Islamic, Solar Hijri, Hebrew, Julian, Gregorian …) converting the text to e g. an ISO 8601 date is not possible.

3.2 Geographic coordinates

RDA supports two cataloguing scenarios: bounding boxes and polygons. The bounding box is described in a fashion similar to how that data is exchanged in MARC 21 and UNIMARC. Further, RDA has instructions for how to record polygons (so far only for maps, not for authorities) in 7.4.3.3: Recording Strings of Coordinate Pairs (RDA 2015, §7.4.3.3). The instruction tells the cataloguer to record the points in the polygon as long/lat and to “list coordinate pairs in clockwise order, starting with the most southeastern vertex of the polygon. [...] The first and last coordinate pairs are the same. [...] If an area or areas within a given polygon are excluded, list the coordinate pairs for any excluded area in counterclockwise order." The treatment of multipolygons is not specified in RDA.

This treatment of polygons is, however, not compatible with the way polygons are described using the geospatial industry-standard WKT. In WKT, a polygon is described as a series of coordinate points (per default long/lat). If a polygon contains several series, the first one is the exterior ring and the other are interior ones (holes). Regarding the order of the coordinate pairs, WKT is precise:

> "The exterior boundary LinearRing defines the 'top' of the surface which is the side of the surface from which the exterior boundary appears to traverse the boundary in a counter clockwise direction. The interior LinearRings will have the opposite orientation, and appear as clockwise when viewed from the 'top'" (OPENGIS 2011, §6.1.11.1 (S.26)).

This means that in RDA the order of the coordinate pairs is exactly the reverse of the order specified in WKT and again, this introduces a tension between the cataloguing code and a possible exchange format.[14]

3.3 So is that a problem?

So are those differences between cataloguing code and exchange formats a problem? It can be, but it does not have to. It is only then a problem when the data is recorded directly into the exchange format (e. g. MARC) without any normalization or standardization. This is the point where cataloguing interfaces enter the equation. A well-designed library system acts as a mediator between the cataloguer entering the data, the machine interfaces where (machine-readable) data exchange takes place and the end-user interfaces where discovery and access take place. That means that the cataloguing interface should allow the cataloguer to describe the resources at hand using local conventions, such as date formats (calendar

[14] This order is not a problem in GML, since the GML representation explicitly specifies if it is an exterior or an interior ring.

used, language-specific names of months), floating point numbers (point or comma as decimal mark) or geographic coordinates (axis order, coordinate reference system) without having to know neither how the system stores this data internally, nor what it looks like when exchanged.

In the data exchange formats, well-known and widely adopted standards must be used wherever possible in order ensure that the information can interoperate with as many external systems as possible. This means that the format specifications need to be precise in both syntax and semantics of the data transported. Some data syntaxes such as XML, Turtle and N-Triples work with datatypes telling a consuming application to interpret a sequence of characters, e. g. “123” as a number by explicitly stating that “123” is an integer as in “123”^^xsd:integer. This can lead to errors, of course, if the data is not semantically correct. In its Linked Data Service, the DNB adds the datatype “xsd:date” to e. g. birth and death dates of persons. When a consumer tried to load this data into a triple store, the software reported several content errors in the data, making it obvious that while “2015-04-31” is a *syntactically* correct date, it is still illegal since there is no April 31 in any given year. We decided to use that as a feature and reported the incorrect dates back to the data maintainers thus implementing a new way to perform quality control.

4 SO WHAT DO LIBRARIES NEED TO DO?

In order to stay interoperable, libraries need to ensure that they use internationally acknowledged, well-known standards when representing their data, so that it can easily be consumed by third-party applications outside of the library domain. Library data does contain much text, e. g. titles or names of persons or corporate bodies. For those, library-specific guidelines for the creation and selection of preferred name forms absolutely make sense, since the use of common, well-documented standards also make it easier to perform string matching and thus to consolidate data from different sources. As soon as the data is not library-specific but we can reasonably expect third-party applications to analyse, convert and mix library data with data from other sources, e. g. in a search engine or a research platform, the use of industry standards must be mandatory.

Librarians also need to actively take part in the work performed by other standards bodies, such as W3C or IETF. The work is often perceived as extremely technical, but in many cases they have working groups that focus on data formats and data exchange, such as the joint W3C/OGC Spatial Data on the Web WG, or the IETF working group for persistent identifiers that also is updating the urn:nbn-schema for the representation of national bibliography numbers as urns.

In order to make data entry easy for cataloguers, system vendors also need pay more attention to how to design user interfaces. A little localisation can go a long way and will probably be a competitive advantage since cataloguing is often externalised. This does not have to be to save costs, but also to allow for more flexible working environments, e. g. a US library with a large Afghanistan collection hiring a cataloguer located in Afghanistan.[15] In this case the cataloguing staff in the USA would have a user interface localised for North America, while the cataloguer in Afghanistan would have a different user interface accepting e. g. dates according to the Islamic calendar. In both cases the ILS converts the data to a common internal format and ensures that the exchanged library data is standards compliant.

After all, books are useless if no one can find them.

[15] Example borrowed from (Düren and Ross 2014).

Acknowledgments

I am very much indebted to Reinhold Heuvelmann and Reinhard Altenhöner for their critical commentary on earlier versions of this paper.

References

Düren, Petra and Rob Ross. 2014. “Risks of Moving to the Cloud: The Human Factor.” Paper presented at: IFLA WLIC 2014 - Lyon - Libraries, Citizens, Societies: Confluence for Knowledge in Session 73 - Information Technology. In: IFLA WLIC 2014, 16-22 August 2014, Lyon, France. http://library.ifla.org/id/eprint/970. Last accessed 2015-06-25.

OpenGIS® Implementation Standard for Geographic information - Simple feature access - Part 1: Common architecture. 2011. portal.opengeospatial.org/files/?artifact_id=25355. Last accessed 2015-06-25.

RDA. 2015. RDA Toolkit. http://access.rdatoolkit.org/. Last accessed 2015-06-25.

UNIMARC Manual. 2009. Authorities Format. Ed. Mirna Willer. 3rd Edition. München: Saur.