Re-broadcasting of bibliographic catalogues in MARC-XML format

Manuel Blázquez-Ochando *

Paper submitted:
May 21, 2013.
Accepted:
August 7, 2013.

Abstract

By using MARC-XML to modify the RSS code format, the technique habitually used by the media to rebroadcast news, bibliographical catalogues can also be re-disseminated. Among other things, this procedure offers the advantages of greater dissemination of collections, the possibility of sharing catalogues with other libraries, and allowing users to read and download catalogues. Researchers performed an array of trials to measure the building and recovery times for such bibliographical collections, while determining the sort of applications and functions needed to control these files. These experiences allow researchers to conclude that it is possible to generate, transmit and retrieve bibliographical catalogues using content syndication practices and methods.

* Universidad Complutense de Madrid, España. manuel.blazquez@pdi.ucm.es

Keywords: Content syndication; MARC-XML; Transmission of bibliographic records; Bibliographic syndication channels.

Resumen

Redifusión de catálogos bibliográficos en MARC-XML
Manuel Blázquez Ochando

La técnica de redifusión de noticias habitualmente empleada en los medios de comunicación social puede ser utilizada en el contexto de los catálogos bibliográficos, modificando su formato de codificación RSS por MARC-XML. Las ventajas que se desprenden de este uso son una mayor difusión de los fondos, la posibilidad de compartir los catálogos con terceras bibliotecas y permitirles a los usuarios la descarga y lectura de éstos. Para lograrlo se han llevado a cabo diversas pruebas que miden el tiempo de creación y recuperación de tales colecciones bibliográficas. Por otro lado se determina qué tipo de programas se necesitan para operar con dichos archivos y cuál es su funcionamiento. Como resultado de estas experiencias se concluye que es factible generar, transmitir y recuperar catálogos bibliográficos mediante técnicas inspiradas en la sindicación de contenidos.

Palabras clave: Redifusión de contenidos; MARC-XML; Transmisión de registros bibliográficos; Canales de sindicación bibliográficos.

Introduction

Online bibliographic catalogues are basic tools in any information and documentation unit. The services commonly offered to the user range from the export of titles consulted for bibliographic management uses, their social labeling, referencing in new scientific works, subsequent consultation, and access to and retrieval of full texts. One of the challenges arising from the ongoing evolution of catalogues is to provide greater dissemination of the catalogues so that these are eventually evenly shared in their entirety with users at no cost. This goal can be achieved by converting bibliographic catalogues to MARC-XML format and their treatment using parser and XML-based structural analyzer programs.

The transfer of bibliographic records via http using MARC-XML files can be based on content syndication or association techniques, as suggested by Blázquez Ochando (2010: 228-392). This doctoral dissertation submits that content syndication techniques, first used in re-broadcasting news in social media, could be used to transmit and recover bibliographic catalogues completely or partially, something that is now underway in the Digital Library of Munich only a year after its publication (Münchener Digitalisierungszentrum Digitale Bibliothek, 2011). Another early experience demonstrating the interest of information centers in adopting MARC-XML as a standard is the doctoral dissertation catalogue initiative of the University of Seville, which allows export and free download of its records in this format (Universidad de Sevilla, 2011).

In the field of Documentation, the most commonly used syndication application consists of the creation of general information channels, the implementation of bibliographic alert services (*ANU Library: new titles*, 2011), re-dissemination of journal articles and contents (Rodríguez Gairín *et al*., 2006) or the grouping of consultations in personalized syndication channels (PUBMED, 2011; Dolan, 2011), where the field of bio-sanitary document management is particularly active.

This paper discusses how to create bibliographical catalogues in MARC-XML format for subsequent retrieval by means of parser programs similar to those used by readers of syndication channels. To verify this process, researchers have provided a trial to show the viability of the transmission of the bibliographic catalogues through the internet and subsequent execution of the programs in the *OrangeUP* platform set up for this purpose (Blázquez Ochando, 2011).

Generation of marc-xml catalogues

Generating catalogues in MARC-XML format (Library of Congress, 2011) requires the availability of bibliographic records in a data base for complete management and treatment. Otherwise, it will be necessary to migrate the information. One method of executing the transfer of the bibliographic catalogue is by exporting it in CSV format, a frequent option used by most bibliographic managers and librarians. For the purposes of this study, we have put together diverse collections ranging from one thousand to one million records from the Library of Congress (Blázquez Ochando, 2010: 299). These collections are shown in *Table 1*.

Table 1. Characteristics of the bibliographic collections tested

Collection	Disc size	Number of records
1000_reg	0.77	1 001
5000_reg	2.68	5 002
10000_reg	5.05	10 004
25000_reg	13.33	25 008
50000_reg	28.34	50 036
100000_reg	54.95	100 054
250000_reg	144.00	250 146
500000_reg	280.49	500 309
1000000_reg	561.39	1 000 039

By taking this step and by means of a PHP-based export program, a MARC-XML catalogue corresponding to the initial bibliographic catalogue can be generated (Blázquez Ochando, 2010: 268-271). To this end, the basic structure of the record is reproduced as are the record node and its dependents as many times as there are tomes and volumes in the collection in question (see *Table 2*).

Table 2. Record model employed

```
<record>

<controlfield tag='001'>Nº Control interno</controlfield>
<controlfield tag='003'>Nº identificación del documento</controlfield>

<datafield tag='017' ind1='' ind2=''>
 <subfield code='a'>Depósito legal o Copyright</subfield>
</datafield>

<datafield tag='020' ind1='' ind2=''>
 <subfield code='a'>ISBN</subfield>
</datafield>

<datafield tag='022' ind1='0' ind2=''>
 <subfield code='a'>ISSN</subfield>
</datafield>

<datafield tag='035' ind1='' ind2=''>
 <subfield code='a'>Número de Control del Sistema</subfield>
</datafield>

<datafield tag='041' ind1='0' ind2=''>
 <subfield code='a'>Código del idioma del documento original</subfield>
</datafield>

<datafield tag='043' ind1='' ind2=''>
 <subfield code='c'>Código geográfico del documento original</subfield>
```

```
</datafield>

<datafield tag='082' ind1='' ind2=''>
 <subfield code='a'>Clasificación Decimal Dewey</subfield>
</datafield>

<datafield tag='100' ind1='1' ind2=''>
 <subfield code='a'>Autor personal</subfield>
</datafield>

<datafield tag='245' ind1='1' ind2=''>
 <subfield code='a'>Área de título</subfield>
 <subfield code='b'>Subtítulo</subfield>
 <subfield code='c'>Mención de responsabilidad</subfield>
</datafield>

<datafield tag='250' ind1='' ind2=''>
 <subfield code='a'>Nº de edición</subfield>
 <subfield code='b'>Mención de edición</subfield>
</datafield>

<datafield tag='260' ind1='' ind2=''>
 <subfield code='a'>Lugar de publicación</subfield>
 <subfield code='b'>Editorial</subfield>
 <subfield code='c'>Año de publicación</subfield>
</datafield>

<datafield tag='300' ind1='' ind2=''>
 <subfield code='a'>Área de descripción física</subfield>
</datafield>

<datafield tag='310' ind1='' ind2=''>
 <subfield code='a'>Periodicidad</subfield>
</datafield>

<datafield tag='490' ind1='0' ind2=''>
 <subfield code='a'>Serie o colección</subfield>
 <subfield code='v'>Nº de serie o colección</subfield>
</datafield>

<datafield tag='500' ind1='' ind2=''>
 <subfield code='a'>Área de notas</subfield>
</datafield>

<datafield tag='654' ind1='0' ind2=''>
 <subfield code='a'>Temática del documento</subfield>
</datafield>

</record>
```

The intervening factor in the process described above is the volume of codifications of the bibliographic records and associated descriptive extension. With regard to the catalogue extension, it should be noted that for collection sizes of 5000+ records the file size is more than 2 MB. This fact, which has also been subsequently contrasted and verified by IndexData (Schafroth, 2010), implies that generation of the corresponding catalogue in a single XML file multiples the size; since it includes characters devoted to its label making treatment, visualization and later retrieval difficult, something that was stated previously (Blázquez Ochando, 2010: 257-258).

The solution to this problem is to create an XML file for each one thousand records, which generally do not surpass 1 Mb in size. This makes file management easier. This approach means that large collections shall require many XML files, which encumbers access to the information in the catalogue. This difficulty can be overcome by employing an OPML file to group the files, as specified for this purpose by Winer (2007). In this way it is possible to retrieve complete catalogue in block (Blázquez Ochando, 2010: 278).

Catalogue retrieval in MARC-XML

The method for MARC-XML format catalogue retrieval entails the use of parser programs capable of analyzing the structure of the XML file and transferring the information for use, whether for display, filtration or retrieval and storage in a data base. This is definitely a process that any aggregator or syndicated reader commonly executes, transposed to the context of bibliographical contents and especially relevant to the field of Documentation.

The example appearing in *Table 3* below is a parser program created in PHP capable of reading and recovering a bibliographic catalogue coded in MARC-XML, such as that shown in *Table 4*. The key to its operation lies in the *simplexml_load_file()* function, available in PHP GROUP (2011a). As specified, this function interprets any XML-based file and converts it into an object that can be accessed in all of its parts by means of DOM (PHP GROUP, 2011b).

Table 3. Field selection model with XPath

```
<?php

$feed = "catalogo.xml";
$xml = simplexml_load_file($feed);

for($i=0; $xml->record[$i]; $i++) {
```

```
    // Campos de control
    $tag001 = $xml->record[$i]->controlfield[0];
    $tag005 = $xml->record[$i]->controlfield[1];

    // Entradas principales
    $tag100a = $xml->record[$i]->datafield[7]->subfield[0];

   // Área de título y mención de responsabilidad
    $tag245a = $xml->record[$i]->datafield[8]->subfield[0];
    $tag245b = $xml->record[$i]->datafield[8]->subfield[1];
    $tag245c = $xml->record[$i]->datafield[8]->subfield[2];

    // Área de publicación
    $tag260a = $xml->record[$i]->datafield[10]->subfield[0];
    $tag260b = $xml->record[$i]->datafield[10]->subfield[1];
    $tag260c = $xml->record[$i]->datafield[10]->subfield[2];

}

?>
```

To verify this end, once the catalogue is loaded in the variable *$xml*, one can simply print to screen, using the *print_r($xml)* function, in order to obtain a result similar to that shown in *Figure 1*.

Table 4. Fragment of recovered data array from MARC-XML bibliographic catalogue

```
SimpleXMLElement Object (
[record] => SimpleXMLElement Object (
[leader] => cabecera[controlfield] => Array (
[0] => número de control
[1] => identificador del número de control
[2] => fecha y hora de la última actualización )
[datafield] => Array (
[0] => SimpleXMLElement Object (
 [@attributes] => Array (
 [tag] => 010
 [ind1] =>
 [ind2] => )
[subfield] => número de control de la biblioteca del congreso )
 [1] => SimpleXMLElement Object (
 [@attributes] => Array ...
```

To retrieve each bibliographic record, one must run all the <*record*> nodes of the MARC-XML catalogue. This task is executed by means of a *for loop* whose execution parameter is exactly the total number of XML file en-

tries to be processed. Within this execution, one can discern how the MARC format encoded labels, stored in variables that have their exact names, are selected. For example, the label *100$a*, which represents the lead author, is stored in variable *$tag100a* and corresponds to the node <*datafield*> placed in position number 7, whose value, in turn, is stored in label <*subfield*>. One can observe that in order to reach the value contained in these labels, the selection route from beginning to end must be indicated, starting with the matrix variable *$xml*, which as has been previously explained contains the entire catalogue content.

Trials with MARC-XML bibliographic catalogues

To confirm the operation of the method of generating and retrieving the MARC-XML format bibliographic catalogues, a sync program was developed (Blázquez Ochando, 2010: 235-310), which allows execution of such operations while providing the execution times and the determining the success or failure of the experiment. The results obtained in *Table 5* demonstrate that the automatic generation of the catalogues takes longer than 15 minutes when the collection in question contains one million records.

Table 5. MARC-XML catalogue creation times

Collection	Time (seconds)
1000_reg	0.24
5000_reg	0.81
10000_reg	1.75
25000_reg	4.82
50000_reg	11.78
100000_reg	26.68
250000_reg	105.28
500000_reg	321.08
1000000_reg	1095.83

Even at that the values obtained with relatively large collections of 50,000 records come in at around 10 seconds. These data stand in contrast to those obtained in the catalogue recovery process. This makes sense because the information transfer operation only requires reading and writing from a known information source, i.e., the data base.

Table 6. MARC-XML Catalogue import times

Collection	Time (seconds)
1000_reg	1.68
5000_reg	8.36
10000_reg	16.85
25000_reg	42.63
50000_reg	92.88
100000_reg	184.64
250000_reg	510.92
500000_reg	1034.99
1000000_reg	2857.61

When the process is reversed, the parser program must read the XML file, generate an object that is accessible in DOM, select the route in which the information is found, display it on screen and, finally, insert it into the data base.

As shown in *Table 6*, this routine considerably increases the execution time and significantly encumbers the work, with times above three minutes, when processing large collections approaching 100,000 records.

Edition and publication test of MARC-XML and RSS catalogues

In order to identify the differences between the development of MARC-XML and RSS format bibliographic catalogues, an online edition and publication test was carried out.[1] Its operation responds to a chain of clearly delimited processes (*Figure 1*).

The *OrangeUP* system has been specifically developed to handle syndication channels and to demonstrate that regardless of format used to describe the bibliographic records or the information contents all of the XML-based formats will have the same transmission, sharing, edition, publication and reading properties. *Figure 1* shows the first steps of the creation of bibliographic catalogues in either MARC-XML or RSS by means of the same method of edition and formularies. Keeping its key code, the bibliographic records can be edited as per the MARC21 bibliographic description standards. Each bibliographic record is assigned a bibliographic syndication

1 See OrangeUP demonstration program available at: http://www.mblazquez.es/testbench/orangeup/

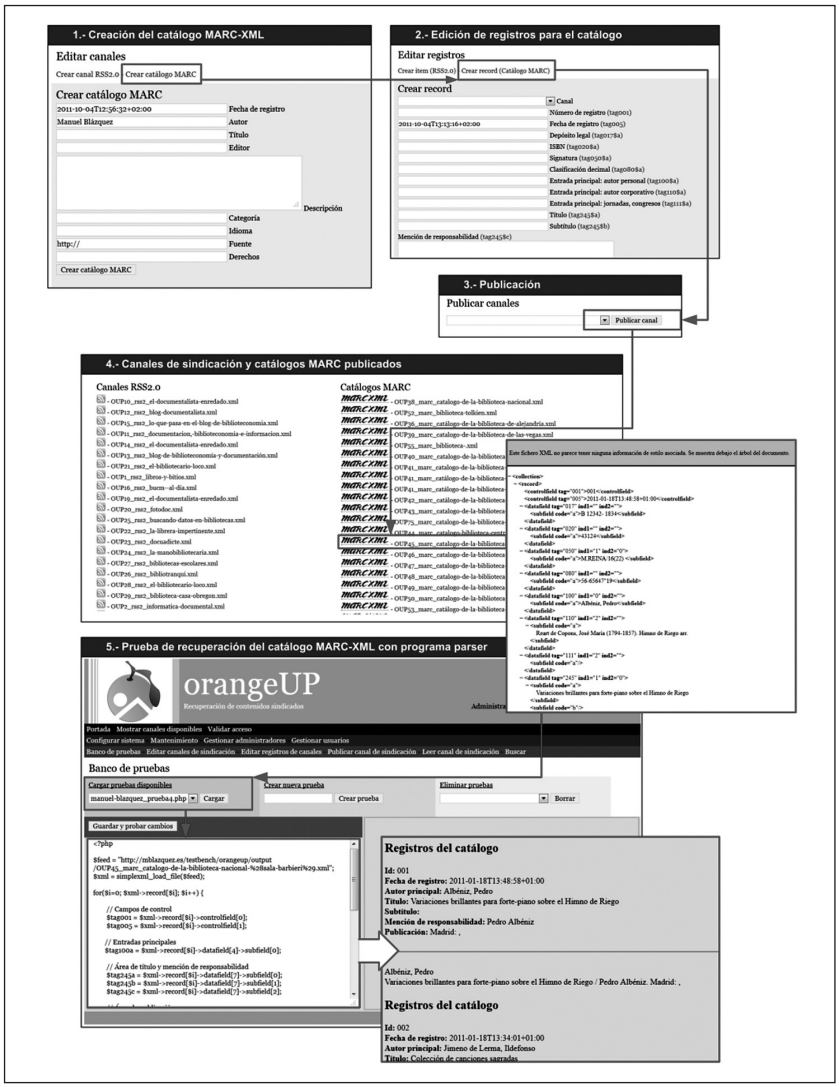

Figure 1. Bibliographic catalogue edition and publication sequence: http://www.mblazquez.es/documents/orangeup-process.jpg

channel.In all events, the records created are practicable and modifiable, i.e., their meta-information, bibliographic fields, category, classification, title areas and mention of party responsible, etc. can be edited. During this process, the program stores the information in the MySQL data base for subsequent publication and encoding in files whose MARC-XML or RSS formats shall be selected by the user.

Subsequently, the registered user can employ the *test bench* function. This is a code editor for testing parser programs, which allows testing of codes such as the one summarized in *Table 3* and executing them in such a way that the results are displayed on the same screen. The result of this process is the complete display of all of the bibliographical records described in the catalogue in the same way the respective items would be displayed by a syndication channel reader. As such, the parallel between the technique of re-dissemination or syndication of contents and the re-dissemination of bibliographical catalogues is undeniable, though there are differences in encoding and evident bias favoring RSS over MARC-XML. To observe the process of edition and publication of the program, see the original video demonstration using *OrangeUp*, available at: http://youtu.be/kS2WiXuRFpM

Conclusions

Bibliographical catalogues can be retrieved in MARC-XML format using parser programs similar to those used to read syndication channels. This allows bibliographic catalogues to be shared between libraries using the methodology previously cited.

Bibliographic catalogue reading and retrieval times are greater than those needed for their creation, because of two key factors: on one hand, the MARC-XML encoding is considerably longer than that for RSS; and, on the other, because of the length of the catalogue document descriptions.

The use of syndication techniques for bibliographic catalogues, as currently in place the Digital Library of Munich, is becoming more and more common. In other cases, export of bibliographic records in MARC-XML format for sharing and reuse by third parties is already a reality. Such is the case of the doctoral dissertation catalogue of the University of Seville. This seems to indicate the initial phase of the implementation of such systems, and a new wave of interest in experimentation in the library and document management milieu.

References

ANU Library: new titles (2011), accessed September 12, 2011, http://anulib.anu.edu.au/about/news/newbooks/

Blázquez Ochando, M. (2010), *Aplicaciones de la sindicación para la gestión de catálogos bibliográficos,* Madrid: Universidad Complutense.

Blázquez Ochando, M. (2011), *OrangeUp*, accessed March 17, 2011, http://mblazquez.es/testbench/orangeup/

Dolan, F. (2011), *MedWorm,* accessed March 15, 2011, http://www.medworm.com/

Library of Congress (2011), *MARC21 XML Schema*, accessed September 17, 2011, http://www.loc.gov/standards/marcxml/

Münchener Digitalisierungszentrum Digitale Bibliothek (2011), accessed September 12, 2011, http://www.digital-collections.de/index.html?c=rss&l=en

PHP GROUP (2011a), *simplexml_load_file,* accessed September 26, 2011, http://php.net/manual/es/function.simplexml-load-file.php

_____ (2011b), *Document Object Model,* accessed September 26, 2011, http://php.net/manual/es/book.dom.php

PUBMED (2011), accessed March 15, 2011, http://www.ncbi.nlm.nih.gov/pubmed

Rodríguez Gairín, J. M. *et al.* (2006), "Sindicación de contenidos en un portal de revistas: Temaria", in *El Profesional de la Información*, 15 (3), pp. 214-221.

Schafroth, D. (2010), *Turbomarc, faster XML for MARC records*, accessed March 18, 2011, https://www.indexdata.com/blog/2010/05/turbomarc-faster-xml-marc-records

Universidad de Sevilla (2011), *Tesis Doctorales: fondos digitalizados,* accessed March 17, 2011, http://fondosdigitales.us.es/tesis/

Winer, D. (2007), *OPML 2.0*, accessed March 17, 2011, http://www.opml.org/spec2