

Rediseño de *OpacMarc* usando conceptos de FRBR

Fernando J. Gómez¹
INMABB (Conicet / Universidad Nacional del Sur)

Resumen: *Litero* es un nuevo OPAC que está siendo desarrollado en el Instituto de Matemática de Bahía Blanca. Surgido como un rediseño de *OpacMarc* en base a las recomendaciones de Martha Yee sobre “FRBRización” de catálogos en línea, *Litero* incorpora mejoras en aspectos de indización, recuperación y visualización, poniendo énfasis en el concepto de obra. Creemos que mediante este OPAC se podrá sacar mayor provecho del potencial que hay en los registros MARC 21/AACR2. Presentamos un prototipo que implementa parte de la funcionalidad deseada, e invitamos a bibliotecarios e informáticos a participar en este proyecto.

OpacMarc

En la Biblioteca Dr. Antonio Monteiro, del Instituto de Matemática de Bahía Blanca (INMABB, dependiente del Conicet y de la Universidad Nacional del Sur), desarrollamos a partir de 2003 un OPAC al que bautizamos *OpacMarc*. En nuestra biblioteca comenzó a usarse para la consulta pública recién en 2006, ya que hasta entonces no disponíamos de suficientes registros en el catálogo automatizado. Ha sido adoptado también por bibliotecas de otras instituciones, entre ellas la UNS, la Asociación Bernardino Rivadavia (Bahía Blanca), el Instituto Balseiro, el Instituto Argentino de Radioastronomía, el Museo Mitre, y el Dipartimento di Fisica, Università degli Studi di Trieste (Italia).

OpacMarc es un software de código abierto², y para desarrollarlo se utilizó la tecnología que mejor conocíamos en aquel entonces: bases CDS/ISIS, el motor WXIS con su lenguaje de programación asociado, y las herramientas CISIS. La principal decisión en cuanto al diseño de *OpacMarc* —y de ahí su nombre— fue la adopción estricta de MARC 21: los datos bibliográficos *debían* estar en ese formato. Eso lo convirtió en el complemento natural para *Catalis*, el otro desarrollo realizado en el INMABB por esos mismos años³. Ambos softwares, junto con *Open MarcoPolo*, han sido integrados en el sistema *CaMPI*⁴.

OpacMarc incluyó algunas características novedosas, en el contexto de los OPACs que teníamos disponibles alrededor de 2003 o 2004:

¹ Av. Alem 1253, B8000CPB Bahía Blanca, Argentina; ffgomez@gmail.com

² <http://code.google.com/p/opacmarc>

³ <http://catalis.uns.edu.ar>

⁴ <http://www.campi.uns.edu.ar>

- acceso mediante índices alfabéticos (no truncados) de nombres, temas y títulos;
- búsquedas por palabras de autor o tema en los índices respectivos (cuando lo usual era hacerlo directamente sobre la base bibliográfica);
- recorrido virtual por los estantes (catálogo topográfico);
- acceso a un esquema de clasificación para realizar búsquedas por tema (Mathematics Subject Classification, para el caso de nuestra colección de Matemática);
- corrección de errores de tipeo (una forma primitiva del “*Did you mean...*” popularizado por Google);
- indización por parte de buscadores, gracias a la posibilidad de recorrer todo el catálogo mediante enlaces de HTML.

No obstante, todavía quedaba un camino por recorrer si queríamos que este OPAC funcionase realmente como un *catálogo*.

Recomendaciones de Martha Yee

El diseñador de un sistema bibliográfico se enfrenta con un problema básico: ¿cómo crear una herramienta simple de usar, capaz de cumplir eficientemente con los objetivos de un catálogo, a partir de una materia prima compleja como son los registros MARC? Martha Yee, del UCLA Film & Television Archive, ha venido publicando recomendaciones para mejorar la indización y la visualización en los OPACs, al menos desde mediados de la década de 1990. Destacamos en particular (Yee & Shatford Layne, 1998) y (Yee, 2003), y los más breves (Yee, 2005) y (Yee, 2006). Se trata de especificaciones detalladas acerca de cómo debiera comportarse un OPAC que aspire a recibir el nombre de *catálogo*, en el sentido de cumplir lo más fielmente posible con los objetivos formulados por Cutter y Lubetzky, que son el pilar —a veces más visible, otras veces menos— sobre el que descansan tanto las AACR2 como FRBR y RDA.

Estas recomendaciones se ocupan de las dos cuestiones centrales de un OPAC:

- Las *búsquedas*, en sus varias facetas: el mecanismo de indización, el diseño de formularios de búsqueda, y el procedimiento para establecer coincidencias (*matchings*) entre los términos provistos por el usuario y los índices. El ejemplo típico: encontrar una obra usando una variante cualquiera del nombre del autor y una variante cualquiera del título. Un elemento clave para el éxito de las búsquedas es el uso efectivo de los registros de autoridad.
- La *presentación (display)* de la información, en particular la presentación organizada de un conjunto de resultados con el fin de facilitar al usuario una rápida comprensión, y de hacer explícitas las relaciones entre personas, obras, manifestaciones, etc. Por ejemplo: presentar las obras asociadas a una persona agrupadas de esta manera: a) las obras creadas por esa persona, b) aquellas

donde esa persona contribuyó de alguna otra forma, y c) aquellas que tratan acerca de esa persona.

El artículo “FRBRization: a method for turning online public finding lists into online public catalogs” (Yee, 2005) estimuló de manera directa el desarrollo de *Litero*. Es importante aclarar que no se trata de una propuesta orientada a implementar el modelo conceptual FRBR en una forma “pura”, libre de las limitaciones que imponen las reglas y formatos preexistentes⁵. La propuesta de Yee, que *Litero* intenta llevar a la práctica, se limita a hacer un uso más efectivo de los datos ya existentes, es decir, nuestros registros MARC 21 creados conforme a AACR2. Se trata de hacer trabajar más intensamente a los datos que tenemos, de manera que el esfuerzo invertido en una catalogación de calidad redunde en un mayor beneficio para los usuarios del catálogo. Lo que Yee llama un catálogo “FRBRizado” es básicamente un catálogo donde las expresiones y manifestaciones de una obra se muestran juntas (de una manera claramente organizada), y separadas de las expresiones y manifestaciones de otras obras.

Identificación de obras en el catálogo

Una limitación de *OpacMarc*, así como de muchos otros OPACs, es la dificultad para acceder de manera precisa y eficiente a una obra, es decir, a una lista que incluya las diferentes versiones o ediciones de una obra en particular, más las de otras obras relacionadas con ella. El problema se nos presentó de manera concreta mientras diseñábamos un servicio de recomendaciones bibliográficas para las materias que se dictan en el Departamento de Matemática de la UNS. Si bien es usual que los docentes citen en los programas de sus materias una edición específica de la obra que recomiendan, en muchos casos no es esa edición particular la que interesa, sino la obra en sí: en cualquier idioma, en cualquier edición. Por ese motivo, queríamos incluir, para cada obra recomendada, un enlace a la página que representa a esa obra en el catálogo. Pero eso requiere que el catálogo maneje alguna noción explícita de “obra”, y *OpacMarc* no lo hace.

El requisito esencial para poder establecer relaciones a nivel de obra entre registros bibliográficos, así como entre éstos y los de autoridad, es que las obras estén adecuadamente identificadas en esos registros. En *Litero* aplicamos el concepto de *identificador de obra* (que es un nombre más útil para lo que en AACR2 se denomina “asiento principal”, o “*main entry*”⁶). Los identificadores de obra pueden desempeñar varias funciones, según cómo y dónde aparezcan dentro de un registro MARC 21. Los siguientes ejemplos, reproducidos de (Yee, 2005), ilustran la variedad de situaciones posibles; el

⁵ Un ejemplo de ese tipo de enfoque es el sistema *Opus*, desarrollado en el CAICYT.

⁶ Para una clara exposición de las razones que justifican la importancia del “asiento principal” (mejor llamado “punto de acceso primario”, o “identificador de obra”) véase el cap. 8 de (Taylor, 2004). Un enfoque algo más teórico puede encontrarse en el cap. 6 de (Svenonius, 2000); allí se usa el término “work ID”.

identificador de obra que nos interesa en el contexto de cada ejemplo se muestra resaltado en negrita.

Identificadores de obras en registros bibliográficos

Caso 1. El registro bibliográfico representa una expresión de la obra identificada

- Identificador de obra **245**, ejemplificado por la obra cinematográfica de Michael Moore:

```
245 00 $a Fahrenheit 9/11 / ...
```

- Identificador de obra **130**, ejemplificado por la obra cinematográfica de 1954 *A star is born* (*Nace una estrella*):

```
130 0# $a Star is born (Motion picture : 1954)  
245 02 $a A star is born / ...
```

- Identificador de obra **1xx más 240**, ejemplificado por una expresión en alemán de la obra de Shakespeare *As you like it*:

```
100 1# $a Shakespeare, William, $d 1564-1616.  
240 10 $a As you like it. $l German  
245 10 $a Wie es euch gefällt / ...
```

- Identificador de obra **1xx más 245**, ejemplificado por una expresión de la obra *Macbeth* de Shakespeare:

```
100 1# $a Shakespeare, William, $d 1564-1616.  
245 00 $a Macbeth
```

Caso 2. El registro bibliográfico representa una compilación de varias obras, incluyendo una expresión/manifestación de la obra en cuestión. En este caso, el identificador de obra se encuentra en un único campo, **7xx con subcampo t** (segundo indicador '2').

- Como ejemplo tenemos una antología que contiene la obra de Arthur Miller *Death of a salesman* (*La muerte de un viajante*):

```
100 0# $a Miller, Arthur, $d 1915-  
245 04 $a The portable Arthur Miller / $c  
edited, and with an introduction by Harold  
Clurman.  
...  
505 0# $a Death of a salesman -- The crucible -  
- Incident at Vichy -- The price.  
700 12 $a Miller, Arthur, $d 1915- $t Death of  
a salesman.  
...
```

Caso 3. El registro bibliográfico representa otra obra que trata *acerca de* la obra en cuestión. También aquí el identificador de obra se encuentra en un único campo, **6xx con subcampo t**.

- Como ejemplo tenemos una antología de ensayos críticos que analizan la obra de Arthur Miller *Death of a salesman*:

245 00 \$a Twentieth century interpretations of
Death of a salesman : \$b a collection of
critical essays / \$c edited by Helena Wickham
Koon.

...

**600 10 \$a Miller, Arthur, \$d 1915- \$t Death of
a salesman.**

Caso 4. El registro bibliográfico representa otra obra que está *relacionada con* la obra en cuestión. También aquí el identificador de obra se encuentra en un único campo, **7xx con subcampo t** (segundo indicador '#').

- Un ejemplo es el registro bibliográfico para una película que es una adaptación de la novela de Margaret Mitchell *Gone with the wind* (*Lo que el viento se llevó*):

130 0# \$a Gone with the wind (Motion picture)
245 00 \$a Gone with the wind / \$c Selznick
International Pictures, Inc. ; in association
with ...

500 ## \$a Based on the novel by Margaret
Mitchell.

...

**700 1# \$a Mitchell, Margaret, \$d 1900-1949. \$t
Gone with the wind.**

- Otro ejemplo es el registro bibliográfico para un disco con la grabación de la banda sonora de la película *Manhattan*, de Woody Allen:

100 1# \$a Gershwin, George, \$d 1898-1937.

240 10 \$a Selections; \$o arr.

245 10 \$a Manhattan \$h [sound recording] : \$b
music from the Woody Allen film ...

...

730 ## \$a Manhattan (Motion picture)

Identificadores de obras en registros de autoridad

En los registros de autoridad hay dos clases de identificadores de obras, y ambos son de un solo campo:

- Identificador de obra **1xx con subcampo t**, ejemplificado por la obra *Cascanueces*, de Tchaikovsky:

```
100 1# $a Tchaikovsky, Peter Ilich, $d 1840-1893. $t Shchelkunchik
```

```
400 1# $a Tchaikovsky, Peter Ilich, $d 1840-1893. $t Cascanueces
```

```
400 1# $a Tchaikovsky, Peter Ilich, $d 1840-1893. $t Casse-noisette
```

```
400 1# $a Tchaikovsky, Peter Ilich, $d 1840-1893. $t Nusscracker
```

```
400 1# $a Tchaikovsky, Peter Ilich, $d 1840-1893. $t Nutcracker
```

- Identificador de obra **130**, ejemplificado por la película *Gone with the wind*:

```
130 #0 $a Gone with the wind (Motion picture)
```

```
430 #0 $a GWTW
```

```
430 #0 $a G.W.T.W.
```

Esta recorrida por los diversos tipos de identificadores de obras nos da una idea de la clase de consideraciones (a veces un tanto sutiles y técnicamente problemáticas, como el hecho de que un identificador de obra a veces consiste de un solo campo, y otras veces de dos) que debieran ser tenidas en cuenta al diseñar un OPAC. Además, puede motivarnos a pensar hasta qué punto le damos importancia al uso de estos identificadores en nuestros propios registros. Y también sirve como ejemplo de una situación donde la comunicación entre catalogadores e informáticos no ha funcionado bien, a juzgar por lo poco representado que está este concepto —tan central para el buen funcionamiento de un catálogo— en la mayoría de los OPACs que conocemos.

Ahora bien, la presencia de estos identificadores es un requisito sobre los *datos* de los que se alimenta un OPAC, y por lo tanto poco puede hacer un OPAC por sí mismo si ese requisito no se cumple de manera satisfactoria. Pero tal vez un OPAC sí puede, de manera indirecta, incentivar a los catalogadores a prestar más atención a ese aspecto de su tarea, si les permite ver que el esfuerzo invertido en incorporar esos datos se manifiesta como un beneficio para los usuarios del catálogo. Yee sostiene que el uso de OPACs que no indizan ni presentan las obras de forma adecuada es una de las causas de que los catalogadores no se ocupen de manera sistemática de la identificación de obras en sus catálogos (Yee, 2005, p. 79).

El prototipo de *Litero*

Luego de una primera etapa de trabajo, entre agosto y noviembre de 2009, contamos con un prototipo que implementa una parte de la funcionalidad deseada⁷. A continuación describimos brevemente las principales características de esta implementación.

Generación de las bases de datos

El proceso de generación de las bases de datos que componen *Litero* toma como entrada un archivo bibliográfico y, opcionalmente, un archivo de autoridades, ambos en formato MARC 21. La salida de este proceso son varias bases de datos:

- La base *bibliográfica*, o de manifestaciones. Ésta contiene esencialmente los datos del archivo bibliográfico utilizado como entrada.
- Una base general de *puntos de acceso*, con un registro por cada punto de acceso presente en la base bibliográfica, y uno por cada referencia de “véase” o “véase además” presente en registros de autoridad asociados a dichos puntos de acceso. Opcionalmente se crean lo que llamamos “puntos de acceso virtuales”, por ejemplo para una entidad principal cuando el catálogo sólo contiene puntos de acceso para algunas de sus entidades subordinadas. En esta base conviven los puntos de acceso para nombres (de personas, de entidades corporativas, geográficos) junto con los puntos de acceso temáticos (encabezamientos de materia o descriptores). Posiblemente también incorporaremos a esa misma base los puntos de acceso para títulos, pero se requiere de más experimentación al respecto.
- Una base de *obras*, que cuenta con un registro por cada identificador de obra presente en la base bibliográfica, y uno por cada referencia de “véase” o “véase además” presente en registros de autoridad asociados a dichos identificadores de obra. Estos registros de obra se representan mediante registros MARC de autoridad. A cada registro de obra se le añaden campos locales con los puntos de acceso (de nombres y temas) presentes en los registros bibliográficos correspondientes a manifestaciones de la obra en cuestión. Esto permite recuperar registros de obras a partir de dichos puntos de acceso, pero se presenta inmediatamente un problema: ¿cómo distinguimos los puntos de acceso de nivel obra de aquellos que corresponden a los niveles expresión o manifestación?
- Bases auxiliares, que facilitan el acceso a índices especializados; éstos se implementan como punteros a subconjuntos de registros de la base de *puntos de acceso* (que es el índice general).

La base de obras desempeña el rol de intermediario entre la base de puntos de acceso y la base bibliográfica. Así, el OPAC presenta *dos capas de*

⁷ Disponible en <http://litero.inmabb-conicet.gob.ar:8080/demo/browse/>

intermediación —puntos de acceso e identificadores de obras— entre las palabras ingresadas por el usuario del catálogo y los registros bibliográficos; es la presencia de estas capas lo que nos permite organizar mejor la recuperación y la presentación de la información bibliográfica.

Indización

Un aspecto que nos interesa resaltar de la indización en *Litero* es el concepto de *sensibilidad jerárquica* (Yee, 2005, p. 83). Podemos ilustrarlo con este ejemplo, donde vemos un registro de autoridad para el Automóvil Club Argentino, y otro para una entidad subordinada:

```
110 2# $a Automóvil Club Argentino
410 2# $a ACA
410 2# $a A.C.A.
```

```
110 2# $a Automóvil Club Argentino. $b Sector de
Cartografía Vial y Turística
410 2# $a Automóvil Club Argentino. $b Cartografía
Vial y Turística
```

La referencia de “véase” en el primer registro nos permite llegar desde “ACA” a “Automóvil Club Argentino”. Pero ¿qué pasa si un usuario busca “ACA cartografía”? Combinando la información de ambos registros, el OPAC debiera llevarlo a “Automóvil Club Argentino. Sector de Cartografía Vial y Turística”; es decir, las referencias de “véase” presentes en un nivel debieran ser aplicadas en todos los niveles que se encuentren por debajo en la jerarquía. *Litero* implementa esta funcionalidad mediante un sencillo mecanismo: al indizar el registro de autoridad de una entidad subordinada E, se incorporan a dicho registro, de manera virtual, las referencias 4xx presentes en los registros de autoridad de todas las entidades jerárquicamente superiores a E. Siguiendo con el ejemplo: al momento de indizar el segundo registro se le incorporan dos campos virtuales heredados del primero (aquí resaltados en negrita):

```
110 2# $a Automóvil Club Argentino. $b Sector de
Cartografía Vial y Turística
410 2# $a Automóvil Club Argentino. $b Cartografía
Vial y Turística
410 2# $a ACA
410 2# $a A.C.A.
```

Para lograr esto, previo a la indización se detectan los vínculos jerárquicos entre puntos de acceso. Esos vínculos también son utilizados para una mejor visualización de los índices de puntos de acceso, por ejemplo al mostrar un encabezamiento de materia junto con sus subdivisiones.

Índices alfabéticos

En *OpacMarc* ya habíamos asignado un rol importante al *browsing* como mecanismo de acceso al catálogo, en base a índices alfabéticos, sistemáticos o topográficos. Inicialmente, en *Litero* nos hemos concentrado en los índices alfabéticos, y hemos implementado el *browsing* sobre un índice general (o integrado) de puntos de acceso, a semejanza del tradicional catálogo diccionario. Una ventaja de este enfoque es que evita al usuario el tener que decidir entre varios tipos de índice, y la consiguiente “penalización” por parte del sistema en caso de haber elegido uno equivocado. (Esa decisión es a veces problemática, incluso para los catalogadores⁸.) *Litero* puede ofrecer además acceso a índices particulares, y en tal caso no estamos limitados a las opciones básicas (autores, temas, títulos), sino que podemos definir índices más específicos, simplemente combinando atributos de los puntos de acceso. Por ejemplo, podemos generar un índice de *intérpretes* (identificados mediante la presencia del código “prf” —*performer*— en un subcampo 4), o uno de *géneros* (campo 655). Otros posibles índices: nombres personales, entidades corporativas, series, personajes de ficción, nombres geográficos, compositores, etc. También podemos generar un índice de *referencias*, donde solamente aparezcan las referencias de “véase” y de “véase además”, así como los puntos de acceso a los que dichas referencias apuntan. Naturalmente este índice no estaría destinado al público, pero podría ser una herramienta útil para un control de calidad del catálogo, o para encontrar rápidamente ejemplos cuando se quiere enseñar a un estudiante qué son y cómo se usan las referencias.

Procesamiento de búsquedas

Podemos dividir las búsquedas que procesa *Litero* en dos grupos: a) búsquedas por palabras clave, usando términos ingresados por el usuario; b) búsquedas por identificadores normalizados —puntos de acceso e identificadores de obras—, usualmente invocadas por el usuario al clicar un enlace. Las primeras aún están siendo desarrolladas; nos hemos dedicado inicialmente a las del segundo tipo, pues como ya dijimos constituyen el fundamento para vincular las diversas entidades que componen el catálogo, y así organizar la presentación de la información. Las búsquedas de obras y de manifestaciones asociadas a un identificador normalizado se descomponen a su vez en varias búsquedas más específicas, con el fin de poder organizar los resultados tal como se muestra en las Figuras 1, 2 y 3.

Interfaz web con Ajax

La denominación “Ajax” se refiere a un conjunto de tecnologías para desarrollo web que permiten modificar ciertas partes de una página HTML, mediante peticiones de datos al servidor que no requieren una recarga

⁸ Véase, por ejemplo, “Name or Subject?”, en el cap. 4 de (Maxwell, 2002).

completa de la página. En el caso de *Litero*, usamos Ajax para producir el efecto de ir abriendo o expandiendo cada uno de los ítems de una lista (puntos de acceso, identificadores de obras, etc.), y así acceder rápidamente a la información asociada con ese ítem, pero sin perder el contexto; este contexto puede ser una porción de un índice alfabético, o una lista de resultados de una búsqueda. No es otra cosa que la metáfora de carpetas y subcarpetas, un mecanismo para explorar estructuras jerárquicas que resulta muy familiar a los usuarios de computadoras. Pero dado que estamos en un catálogo, podemos también pensar en una imagen de otros tiempos: el catálogo de fichas. El catálogo de fichas era —a la fuerza— una invitación a recorrer la colección. La ficha que nos interesaba estaba siempre en un contexto, rodeada posiblemente de otras que podían tener alguna relevancia para nuestra búsqueda. Y si la ficha buscada no estaba, alguna de las vecinas podía sernos de utilidad. Mucho de eso se perdió al pasar a los catálogos en línea, donde los índices no siempre están disponibles, y tantas veces se nos obliga a ingresar palabras en un formulario como única vía para asomarnos a la colección⁹.

Es importante señalar que *Litero* no dependerá de Ajax: todo el contenido del catálogo debe estar disponible para los usuarios cualquiera sea su navegador, así como para los buscadores web.

El diseño de la interfaz presenta problemas aún no resueltos. Algunos son inherentes al uso de Ajax, como la pérdida del estado de una página en relación con el uso del botón “Atrás”. Otros tienen que ver con el mecanismo de “visualización jerárquica” que acabamos de describir: ¿cómo evitar que esa modalidad de acceso a la información —ciertamente útil en casos de autores prolíficos u obras con gran cantidad de versiones— se convierta en una molestia en los casos simples, es decir, aquellas obras que están representadas en el catálogo por una única manifestación?

Organización de listas de obras y manifestaciones

En *Litero* aplicamos un sencillo principio de organización a los resultados asociados a un punto de acceso: agrupamos una lista de obras asociadas a un autor según el tipo de relación entre cada obra y dicho autor (Figura 2); análogamente, agrupamos una lista de manifestaciones asociadas a una obra según el tipo de relación entre cada manifestación y dicha obra (Figura 3). En las Figuras 2 y 3 vemos además otra característica que hace a la usabilidad del catálogo: la presentación *comprimida* de listas largas, para que el usuario pueda tener una visión general del conjunto recuperado.

⁹ Es cierto que el contexto alfabético puede resultar, en muchos casos, de poca relevancia. El usuario puede estar más interesado en el contexto de “escritores colombianos” o “novelistas latinoamericanos” que en el contexto de “personas llamadas García”, pero creemos que tener un contexto —si responde a una lógica sencilla— es mejor que no tener ninguno. De hecho, algunos puntos de acceso alfabéticamente cercanos pueden ser relevantes, aunque sólo sea debido a la presencia en el catálogo de variantes (erróneas o no) del nombre buscado.

☐ Hofstadter, Douglas R., 1945-

☐ 7 obras de Douglas R. Hofstadter

Hofstadter, Douglas R., 1945-

1. ☐ **Ambigrams.** [1]
2. ☐ **Fluid concepts & creative analogies** [1]
3. ☐ **Gödel, Escher, Bach** [3]
4. ☐ **I am a strange loop** [1]
5. ☐ **Metamagical themas** [1]
6. ☐ **Mind's I** [1]
7. ☐ **Ton beau de Marot** [1]

☐ 4 obras relacionadas con Douglas R. Hofstadter

Cope, David, 1941-

1. ☐ **Virtual music** [1]

Nagel, Ernest, 1901-

2. ☐ **Gödel's proof** [1]

Pushkin, Aleksandr Sergeevich, 1799-1837.

3. ☐ **Evgenii Onegin.** [2]

Veltroni, Walter, 1955-

4. ☐ **Scoperta dell'alba.** [1]

1.  The discovery of dawn / Walter Veltroni ; [translated and with a preface by Douglas Hofstadter]. — New York : Rizzoli Ex Libris ; Enfield : Publishers Group UK [distributor], 2008. — vi, 140 p. ; 22 cm.	PQ4882.E446 S3613 2008
--	-----------------------------------

☐ 1 obra acerca de Douglas R. Hofstadter

Palleschi, Aldo.

1. ☐ **Contrapposizione e isomorfismo** [1]

Figura 1. Visualización de la lista de obras asociadas con el punto de acceso para Douglas Hofstadter. Para una de ellas, se muestra en un formato breve el único registro bibliográfico asociado.

[-] **Shakespeare, William, 1564-1616**

- + 42 obras de *William Shakespeare*
- + 11 obras relacionadas con *William Shakespeare*
- + 2 obras acerca de *William Shakespeare*

Figura 2. Visualización comprimida de las obras asociadas al punto de acceso *Shakespeare, William, 1564-1616*.

- 3. + **Hamlet.**
- 4. + **Julius Caesar.**
- 5. [-] **Macbeth**
 - + 52 ediciones de *Macbeth*
 - + 2 obras que contienen *Macbeth*
 - + 46 obras relacionadas con *Macbeth*
 - + 16 obras acerca de *Macbeth*

Figura 3. Visualización comprimida de las manifestaciones asociadas al identificador de obra *Shakespeare, William, 1564-1616. Macbeth*.

Selección de datos de prueba

Para poder ir probando las funcionalidades de un OPAC es necesario contar con una muestra de registros bibliográficos y de autoridad que, sin ser demasiado voluminosa, sea representativa de la variedad de situaciones que nos interesa considerar. Para el desarrollo de *Litero* hemos tomado principalmente registros del catálogo de Library of Congress. Al momento de escribir este trabajo, tenemos 342 registros bibliográficos y 27 de autoridad. Algunos han sido seleccionados por corresponder a ejemplos incluidos por M. Yee en sus trabajos; otros tienen que ver con personas o entidades de Argentina, a fin de darle algo de sabor local al prototipo. Hasta ahora estos registros han sido usados para testeos manuales. En una próxima etapa, deberían ser utilizados también para testeos automáticos. En efecto, nos interesa explorar métodos de validación de software que nos permitan tener la certeza de que *Litero* funciona correctamente.

Herramientas usadas

El desarrollo de *Litero* fue encarado con mejores herramientas informáticas que las utilizadas para *OpacMarc*. La tabla sintetiza las principales diferencias:

	<i>OpacMarc</i>	<i>Litero</i>
Programación en el servidor	WXIS	Python
Framework para desarrollo web	(Ninguno)	Django
Programación en el cliente (JavaScript)	(Uso escaso)	Ext JS, Ajax
Bases de datos	CDS/ISIS	Malete ¹⁰

No vamos a entrar en detalles sobre las cuestiones informáticas; nos limitaremos a indicar que esta renovación de herramientas trae consigo varias ventajas. Entre ellas se destacan: el uso de programación orientada objetos; la aplicación del modelo MVC, que permite organizar de una manera muy limpia la estructura de una aplicación web; la posibilidad de utilizar Unicode en los datos del catálogo; y las facilidades para la internacionalización del software (i.e., la traducción de la interfaz a varios idiomas).

Documentación

Otro cambio importante tiene que ver con la forma de encarar la documentación del software. *Litero* está siendo desarrollado con una técnica de programación conocida como *literate programming* (Knuth, 1984), nombre que podría traducirse como “programación letrada”, o “programación literaria”. Se trata de una forma de escribir software en la que documentación y código se entrelazan; más aun, es la documentación la que guía el proceso de escritura. La idea central es que en vez de escribir el código de un programa en el orden en que la computadora lo requiere, escribimos un texto donde explicamos cómo hacemos el programa; y organizamos las secciones de ese texto en el orden que nos parece más conveniente para que un lector lo entienda. Luego, de ese texto se extraen automáticamente tanto el código del programa como la documentación del programa. Así, con *Litero* tendremos un OPAC y también una explicación detallada de cómo funciona internamente ese OPAC.

Además del valor evidente que tiene una buena documentación para facilitar la comprensión y el mantenimiento de un software por parte de los programadores, creemos que en el caso de un OPAC hay una necesidad particular de buena documentación, con el fin de reducir la brecha comunicacional entre bibliotecarios e informáticos. No son pocos los bibliotecarios que, sin necesidad de ser programadores, están en condiciones de comprender una explicación del funcionamiento de un software, y a través

¹⁰ Se trata del motor de bases de datos conocido originalmente como OpenIis. Disponible en <http://malete.org>.

de esa explicación detectar que algo no está del todo bien, sugerir mejoras, y en definitiva, tener un rol más activo en la creación del propio catálogo¹¹. Esperamos que *Litero* sea también una contribución en este aspecto.

Conclusiones y próximos pasos

Quizás la principal conclusión que puede extraerse de esta experiencia inicial con *Litero* es que, aun sin recurrir a nuevas reglas o nuevos formatos, los registros MARC 21 creados en base a AACR2 contienen un potencial que ha sido escasamente aprovechado. En realidad, no era necesario crear un nuevo OPAC para advertir este hecho, pero ciertamente es más sencillo comunicarlo si se lo puede mostrar “en vivo”.

Por otra parte, teniendo en cuenta la inminente publicación de RDA, y con ella la incorporación explícita de conceptos de FRBR al trabajo de catalogación, se abren perspectivas promisorias para continuar explorando formas de mejorar nuestros catálogos. En este sentido, *Litero* podría ser de utilidad para los catalogadores en la etapa de transición que se avecina, pues les permitirá ver que, al fin y al cabo, algo de FRBR ya había en nuestros catálogos, sólo que lo teníamos un poco escondido.

Los siguientes pasos en este desarrollo estarán guiados principalmente por la lista de problemas pendientes y de funcionalidades que aún falta implementar (entre ellas, todas las que ya existían en *OpacMarc*), pero también por la respuesta que obtengamos a partir de la presentación pública de *Litero*. Nos interesa darle a este software una temprana y amplia exposición pública, con el propósito de someterlo a testeos variados, detectar problemas, probar soluciones alternativas, y en general mejorarlo con la colaboración de los interesados. Son por lo tanto bienvenidos tanto bibliotecarios como informáticos que deseen aportar opiniones, experimentar con sus propios datos, y contribuir a la mejora de *Litero*.

Agradecimientos

Deseo expresar mi reconocimiento a las autoridades del INMABB por su apoyo. Agradezco a los organizadores del II Encuentro Nacional de Catalogadores, cuya convocatoria actuó como disparador de este desarrollo. Finalmente, gracias a Martha Yee por el estímulo: primero indirectamente a través de sus textos, y luego respondiendo mis consultas por correo electrónico (pero desde luego, ¡ninguna de las fallas actuales o futuras de *Litero* deberán ser atribuidas a ella!).

¹¹ J. McRee Elrod escribió a mediados de los '90, en su artículo *Cataloguer's role in catalogue construction - A modest proposal*: “There was a time when we as cataloguers were concerned with building catalogues. Now we seem primarily to be concerned with creating bibliographic records, leaving catalogue construction to others.” (Recuperado de <http://www.slc.bc.ca/mac/catcon.htm>)

Bibliografía

Knuth, D. (1984). Literate programming. *The Computer Journal*, 27, 97-111. Reproducido en: D. Knuth, *Literate programming* (pp. 99-136). Stanford, California: Center for the Study of Language and Information, 1992. Disponible también en <http://www.literateprogramming.com/knuthweb.pdf>.

Maxwell, R. (2002). *Maxwell's guide to authority work*. Chicago: American Library Association.

Svenonius, E. (2000). *The intellectual foundation of information organization*. Cambridge, MA: MIT Press.

Taylor, A. (2004). *The organization of information* (2nd ed.). Westport, CT: Libraries Unlimited.

Yee, M., & Shatford Layne, S. (1998). *Improving online public access catalogs*. Chicago: American Library Association.

Yee, M. (2003). *Principles for the display of cataloger-created metadata*. Disponible en <http://slc.bc.ca/yee.pdf>.

Yee, M. (2005). FRBRization: a method for turning online public finding lists into online public catalogs. *Information Technology and Libraries*, 24(3), 77-95. Postprint disponible en <http://repositories.cdlib.org/postprints/715>.

Yee, M. (2006). Applying FRBR to library catalogues: A review of existing FRBRization projects. Disponible en http://www.nla.gov.au/lis/stndrds/grps/acoc/documents/Yee_FRBR.doc.