4-17-2013

Keeping up with Ebooks: Automated Normalization and Access Checking with Normac

Kathryn Lybarger
University of Kentucky, kathryn.lybarger@uky.edu

code{4}lib JOURNAL

Issue 20, 2013-04-17

ISSN 1940-5758

Keeping up with Ebooks: Automated Normalization and Access Checking with Normac

Cataloging ebooks is difficult to do well, as they are often purchased in large collections, sometimes with only low-quality cataloging copy available. MARC records may be provided upfront in a large batch, or trickle in one at a time as they become available. Records may contain links that point nowhere, to the wrong book, or to an offer to sell you the book you already own. Loading records sight unseen may introduce inconsistency or overlay good print records with poor electronic ones, making the catalog much more difficult to search.

This article describes in more detail the major challenges in ebook cataloging, record normalization and access checking, and introduces Normac: an open source web-based tool for processing MARC records.

by Kathryn Lybarger

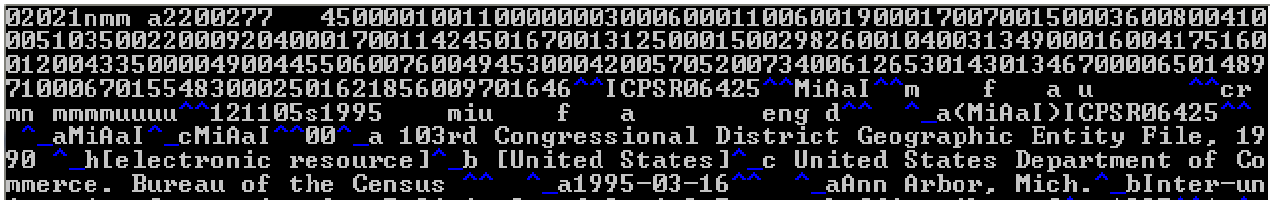
Introduction

Cataloging ebooks is difficult to do well, as they are often purchased in large collections, sometimes with only low-quality cataloging copy available. MARC records may be provided upfront in a large batch, or trickle in one at a time as they become available. Records may contain links that point nowhere, to the wrong book, or to an offer to sell you the book you already own. Loading records sight unseen may introduce inconsistency or overlay good print records with poor electronic ones, making the catalog much more difficult to search.

Traditional tools for dealing with MARC may not easily handle the challenges of large batches with complicated conditional field edits, especially when faced with multiple cataloging standards present in a single batch. Checking not only that a link points somewhere, but that it points to the book in question is also beyond the standard features offered by cataloging software or web site link checkers. To help solve these problems, I wrote Normac: a new software package for editing batches of MARC records to bring them up to local standards and checking ebook access. Once the system is configured, a new batch of MARC records can be quickly processed through a web interface or from the command line.

Some background – MARC Editing

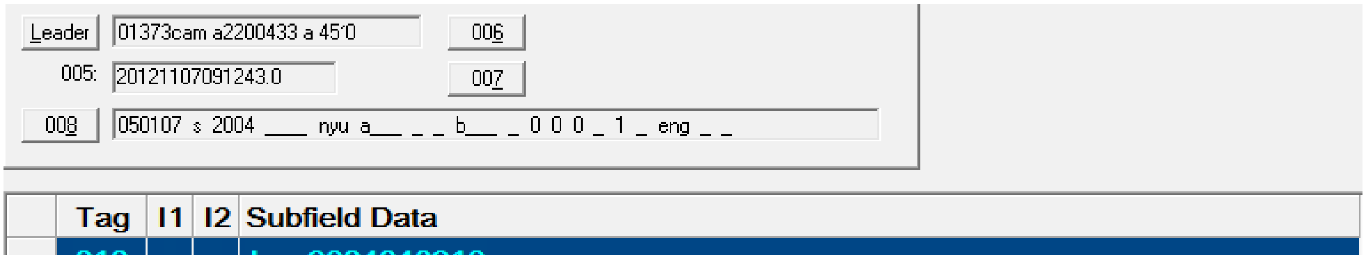
MARC is a flexible file format for data encoding and interchange, but in its native binary format it is not easy to edit:



```
02021nmm a2200277   4500000100110000000300060001100600190001700700150003600800410
005103500220009204000170011424501670013125000150029826001040031349000160041751 60
012004335000049004455060076004945300042005705200734006126530143013467000006501489
71000670155483000250162185600970l646^^ICPSR06425^^MiAaI^^m     f    a u        ^^cr
mn mmmmuuuu^^121105s1995     miu     f    a          eng d^^  ^_a(MiAaI)ICPSR06425^^
 ^_aMiAaI^_cMiAaI^^00^_a 103rd Congressional District Geographic Entity File, 19
90 ^_h[electronic resource]^_b [United States]^_c United States Department of Co
mmerce. Bureau of the Census ^^  ^_a1995-03-16^^  ^_aAnn Arbor, Mich.^_bInter-un
```

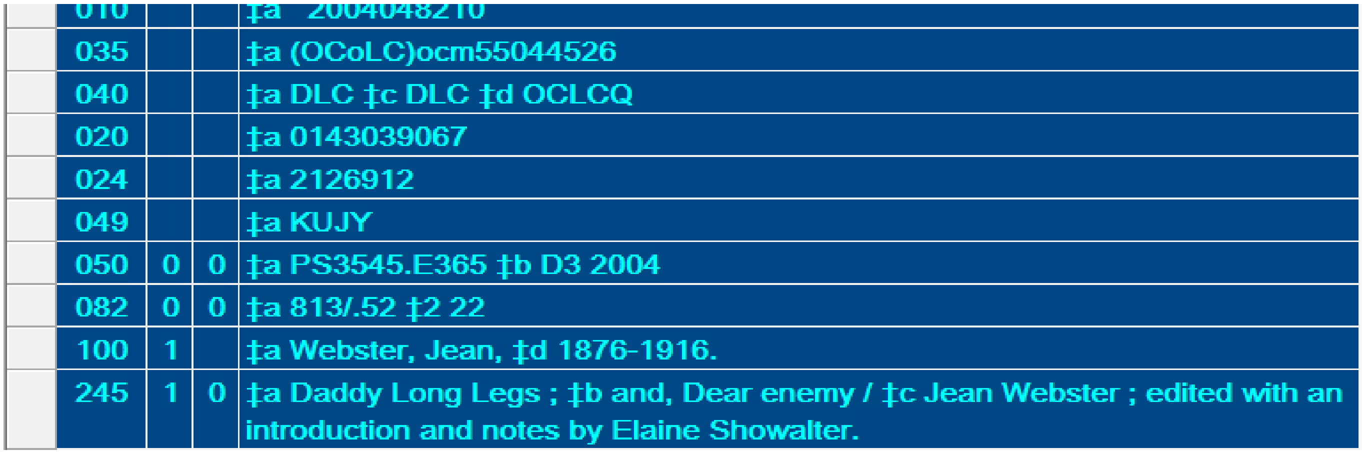
Figure 1: Marc Record in its Native Format

Most ILS and cataloging software (such as Ex Libris Voyager) include a friendly editor for record-by-record MARC editing:



```
Leader  01373cam a2200433 a 45 0
005:    20121107091243.0
008     050107 s 2004 ____ nyu a___ _ _ b___ _ 0 0 0 _ 1 _ eng _ _
```

	Tag	I1	I2	Subfield Data



	010			‡a 2004048210
	035			‡a (OCoLC)ocm55044526
	040			‡a DLC ‡c DLC ‡d OCLCQ
	020			‡a 0143039067
	024			‡a 2126912
	049			‡a KUJY
	050	0	0	‡a PS3545.E365 ‡b D3 2004
	082	0	0	‡a 813/.52 ‡2 22
	100	1		‡a Webster, Jean, ‡d 1876-1916.
	245	1	0	‡a Daddy Long Legs ; ‡b and, Dear enemy / ‡c Jean Webster ; edited with an introduction and notes by Elaine Showalter.

Figure 2: ILS Marc Editor

Many also include batch editing functionality, such as Ex Libris's Global Data Change and Millennium's Global Update.

There are also software libraries available for manipulating MARC records, including the python library PyMARC [1] and the Perl library MARC/Perl [2].

Perhaps most popular among librarians is Terry Reese's software MarcEdit [3]. It provides batch conversion of MARC records between several formats, including binary MARC, mnemonic text MARC and MARCXML, and does character set conversion between MARC-8 and Unicode. It also includes a powerful yet friendly MARC Editor for applying common edits (including regular expressions) to mnemonic MARC files. MarcEdit also includes a GUI script wizard which creates VBScript programs to perform common operations; those scripts may be edited to perform arbitrary operations on MARC batches.

Challenges in Normalization

When adding ebook records to our catalog, we would like to do so efficiently and expediently, with consistent quality in the records, and only including working links.

One challenge stems from the variability in size of record batches. Often when we add a new collection to the catalog, MARC records for all titles are available upfront from the vendor. We may have thousands of records to deal with at once, which is not itself a problem; MarcEdit can handle large numbers of records, and many powerful checks and modifications can be done through its MarcEditor interface. That is an efficient way to handle large batches (spending relatively little time per record) but the efficiency falls off if we do that same MarcEditor procedure frequently on smaller batches of files, such as an update to that collection. MarcEdit addresses this need in some instances with its Script Wizard add-in, which allows some of its more common functions to be saved as a script, but more complex edits may be desired. It may be quicker to modify small batches one record at a time, but this sacrifices consistency. It may be more efficient to wait until we have a larger batch of titles before performing the batch edits, but we prefer to make the titles available to our patrons as soon as possible. It would be nice to perform complex edits to batches of MARC records efficiently and with consistent results regardless of batch size.

Another challenge stems from the variability of cataloging standards that may be present in a single batch of records. Records for a single collection gathered from a collective like OCLC may have been created at different times by different catalogers following different standards. Some cataloger judgment is allowed in what makes a record acceptable quality: records may vary in their inclusion of call numbers, authorized subject headings, and contents notes. Some records even contain content specific to an individual institution (and not useful to others) such as URLs modified to restrict access with a proxy prefix.

Records in a collection may vary not only in details, but also may just use different codified standards. Prior to 2009, ebooks were cataloged as reproductions of print books, with each record specific to the vendor who had "digitized" it (even if the book was born digital). Most fields in the record were identical to those in the corresponding print record, with a few fields and subfields to indicate its electronic nature:

```
245 04 $a The Developer's guide to debugging $h [electronic resource] / $c by Thorsten
Gro?tker, Ulrich Holtmann, Holger Keding, Markus Wloka.
300 __ $a xix, 224 p. : $b ill. ; $c 23 cm.
533 __ $a Electronic reproduction. $b New York : $c Springer, $d 2008. $f (Computer Science).
```

```
$n Mode of access: World Wide Web. $n System requirements: Web browser. $n Access may be
restricted to users at subscribing institutions.
```

In 2009, the recommended standard changed to provider-neutral records [4]; that is, one record describes all potential electronic versions of a title and includes links to all known versions. This change seemed to raise the overall quality of ebook records (since all improvements to a title's metadata were concentrated on one record), but the new standard brought other issues with it as well. Notes that would have been included with provider-specific records, such as:

```
538 __ $a Mode of access: World Wide Web.
506 __ $a Restricted to subscribers.
```

were no longer appropriate, because they would not apply to an open access version available via FTP. The format of the physical description is also different:

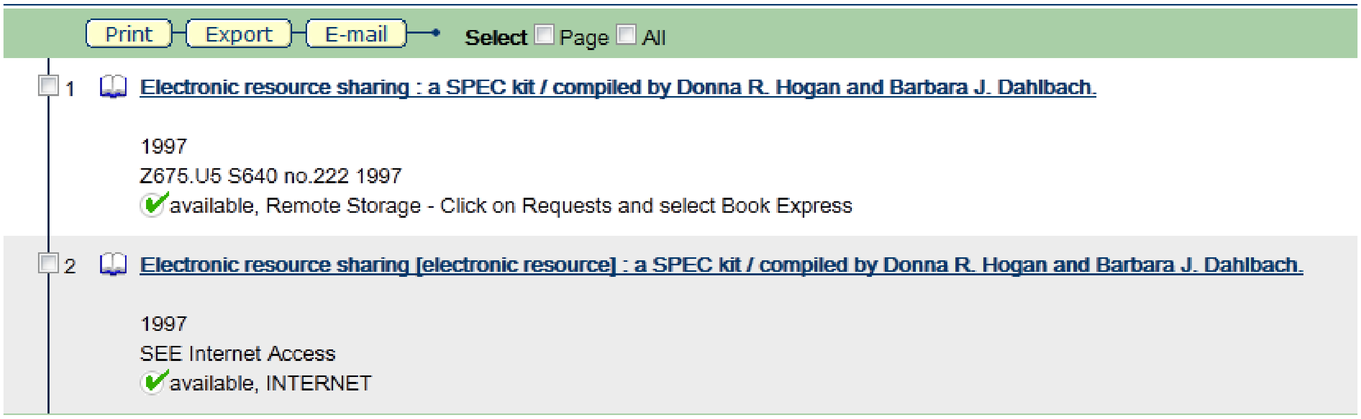
```
300 __ $a 1 online resource (xix, 224 p.) : $b ill.
```

and added entries for ebook collections or vendors that would previously have aided the cataloger to collocate all ebooks in a package would be removed. Patrons using your catalog may not notice these differences, but they would certainly be confused by the appearance of multiple links (one for each vendor), most of which did not work for them. For consistency and utility, catalogers may want to standardize formats, restore absent notes, and remove inappropriate links.

Again, in 2013, we are seeing a mix of standards in record batches as some cataloging agencies are switching to the new cataloging standard RDA (Resource Description and Access) [5] while others are continuing to catalog in AACR2. Many differences between the two standards do not necessitate different processing procedures, though a couple of them might, depending on your local cataloging environment. For example, RDA records include three new physical description fields (33X), indicating the content, media and carrier types of the resource:

```
245 __ $a For whom the bell tolls.
336 __ $a two-dimensional moving image $2 rdacontent
337 __ $a computer $2 rdamedia
338 __ $a online resource $2 rdacarrier
```

However, many ILS's are not prepared to handle these new fields. While it may be straightforward enough to add 33X fields to the tag table so they don't cause errors when used, it is quite another to make the ILS display them in a way that is as clear to patrons as the GMD:



□ 1 Electronic resource sharing : a SPEC kit / compiled by Donna R. Hogan and Barbara J. Dahlbach.

1997
Z675.U5 S640 no.222 1997
✓available, Remote Storage - Click on Requests and select Book Express

□ 2 Electronic resource sharing [electronic resource] : a SPEC kit / compiled by Donna R. Hogan and Barbara J. Dahlbach.

1997
SEE Internet Access
✓available, INTERNET

Figure 3: Opac Record Displaying GMD

Until their ILS software catches up with the change in standards, libraries are handling RDA records in various ways. Some are loading them as-is, but others are making changes such as deleting the 33X fields, applying a static GMD to batches known to be of a certain type (electronic resource, video recording, etc) or copying data from 33X fields to form a new GMD, such as:

```
245 __ $a For whom the bell tolls $h [two-dimensional moving image : electronic resource]
```

With a mix of record standards and quality, the cataloger doing batch editing may want to apply unusual logic, such as “If the record is coded RDA in the 040, and there is no 245$h, concatenate the first instances of 336 and 338 and add that string as a 245$h following its $a.”

Such needs are unlikely to go away over time. When ILS software can handle these new fields effectively, catalogers may wish to add them even to AACR2 records; OCLC is already allowing such changes to its master records [6].

Using Normac’s normalizer, you can write a simple configuration file to perform common functions (such as adding and deleting fields) and run pre-defined functions to perform more complicated functions (such as converting a record to the provider-neutral standard, or moving OCLC numbers to a 776 field). You can also write (and share) arbitrarily complicated functions to be run on your MARC records.

Challenges in Access Checking

There are many popular solutions for the problem of web site access checking, including the W3C Link Checker [7] and Xenu’s Link Sleuth [8]. Unfortunately, many link checkers work by checking HTTP status codes [9]:

Some common status codes:

```
200 - OK - request successful
403 - Forbidden - server refuses to fulfill your request
404 - Not Found - server could not find anything matching your request
```

If the software queries the web server with the URL and receives a good code like 200, it will consider the link to be good. If it finds a bad code like 403 or 404, it will report the link as bad. When the web pages in question are ebooks, the answers are not that simple.

An ebook may appear fine to a link checker, even when you don’t really have access to its content. Many ebook links point to a metadata page with a table of contents; that page is a valid web page even if you don’t have access to individual chapters. If you have your link checker go to a greater depth (not only check the link, but the links on the page it returns), some chapters such as front matter may be available and others may not. Even if the content is not available to you, the link checker can still be fooled; rather than throwing up an unfriendly 404 page, the site may redirect you to a web form where you can login with different credentials or purchase the content, and that web form may be a valid web page with a 200 status.

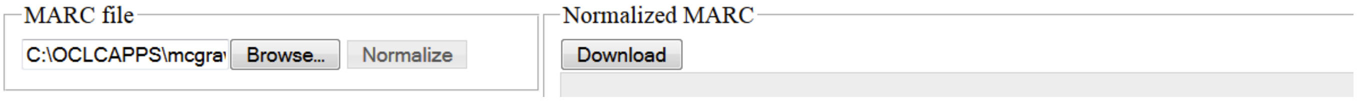
Similarly, an ebook may appear bad to a link checker when it is just fine: a 403 status may be returned for all links if the ebook platform just does not allow link checking by search engine robots. An ebook may instead have a broken link which is relatively easy to fix. Many ebooks have DOIs, which are great for inclusion in ebook records; in the case of a platform migration or other change, the vendor only needs to register the change with CrossRef to effect the change in all catalogs at once. If the DOI is not registered properly however, the catalog link will return a page indicating a bad DOI. When this situation is detected, you can just replace the DOI with the direct link in your catalog, but it is preferable to report the DOI as broken, fixing the broken link for libraries that have already loaded the record with the broken link.

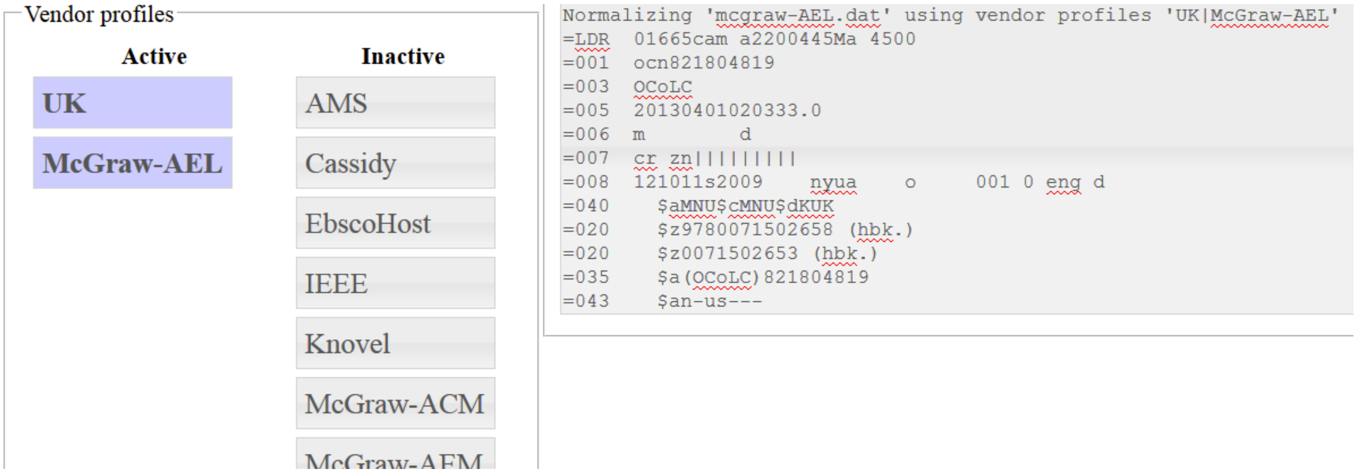
Using Normac’s access checker, you can check a list of ebooks for actual functional access based on the specific ebook platform. With some programming, the software may be configured to search for key phrases from that platform (such as “not available to your institution”) or perform arbitrarily complex actions, such as searching the table of contents for the first non-frontmatter chapter and confirming that the page returned by that link is in PDF format. After all checks are done for a batch, different types of access errors (no vendor access, broken DOI) are identified for reporting to the appropriate vendors.

Normac

Normac is web-based software that normalizes MARC records according to institutional policy and vendor specifics, and then verifies access to any linked online resources. It can be easily configured to add, delete and modify fields in common ways, and can perform arbitrarily complex checks and edits with additional programming.

Once Normac is installed and configured, you can process MARC batches from the command line, or by feeding them to a web page and selecting settings for the batch. Normalization and customization finish quickly, prompting only for edits that require manual intervention, and then the modified file is made available for download.


MARC file
C:\OCLCAPPS\mcgra



```
Normalizing 'mcgraw-AEL.dat' using vendor profiles 'UK|McGraw-AEL'
=LDR  01665cam a2200445Ma 4500
=001  ocn821804819
=003  OCoLC
=005  20130401020333.0
=006  m        d
=007  cr zn|||||||||
=008  121011s2009    nyua    o      001 0 eng d
=040    $aMNU$cMNU$dKUK
=020    $z9780071502658 (hbk.)
=020    $z0071502653 (hbk.)
=035    $a(OCoLC)821804819
=043    $an-us---
```

Figure 4: Normac

If you would also like to do access checking for links in the MARC record batch, you can then submit the modified file to the access checking queue. If you wish to check access for a collection already in your catalog, you can also just submit a list of those links. Link checking is a slower process, mainly for consideration to the vendor; requests will only be sent to a given vendor approximately every ten seconds so as not to be a nuisance.

After links are checked, the MARC batch is split into two files: one with working links and one where links have problems. A report is generated indicating the types of errors, such as broken DOIs, deleted resources or lack of institutional access. The good file may be then loaded into the catalog, while problems are investigated. After problems are addressed, the problem file may be re-submitted for link checking before loading.

Customizing Normalization for Institutions and Vendors

To simplify creation of workflows for new vendors, Normac accepts vendor profiles describing changes that should be made to records from that vendor. This profile includes static fields that should be added, patterns of fields that should be deleted, and functions that should be applied to the record at various times. Profiles are encoded in an ini-style file (example: UK.ini) with functions implemented in a sibling PHP file (example: UK.php).

Multiple profiles can be applied to a batch, allowing an institution's main rules to be encoded in a primary profile and having smaller, simpler profiles specific to individual vendors. For example, an institution's profile may have the form:

```
[Description]
Base behavior for ebooks processed at UK; additional behavior should be added to individual
vendor profiles

[Variables]
ProxyPrefix = "http://ezproxy.uky.edu/login?url="
PublicNote = " -- CLICK HERE for Internet Access to title"
GoodGMD = "electronic resource", "videorecording"

[Initial]

[Delete]
=029
=506
=533
=538
=6.. .[^02]
=938

[Middle]
RemoveInvalidLinks
Add856Label
```

```
Add856ProxyPrefix
Add856PublicNote
Neutralize300
AddGMD

[Add]
=099 \\$aSEE Internet Access
=538 \\$aMode of access: World Wide Web.
=506 \\$aRestricted to subscribers.

[Final]
```

That institution's profile for Wiley ebooks might look like:

```
[Description]
Extra processing for Wiley ebooks

[Variables]
ValidUrlHosts = dx.doi.org/10.1002, onlinelibrary.wiley.com
Label856 = Wiley Online Library

[Initial]
SpecialWileyStuff

[Delete]
=710.*Wiley InterScience

[Middle]

[Add]
=730 0\$aWiley Online Library.

[Final]
```

With these two profiles defined and applied (in the order UK, Wiley), the following occurs:

1. UK profile Variables are set.
2. Wiley profile's Variables are set, possibly overwriting UK's
3. UK's Initial functions are run.
4. Wiley's Initial functions are run.
5. UK's Delete lines are deleted.
6. Wiley's Delete lines are deleted.

… (and so on)

Given the modular nature of vendor profiles and functions defined in them, code for small specific tasks can be compartmentalized, allowing them to be quickly re-ordered within a procedure, easily modified, and shared between institutions.

Customizing Access Checking for Platforms

Normac accepts platform profiles that distinguish between a good ebook link and bad ones of various types. This profile is a PHP file containing a function valid_url that accepts a URL and returns 0 for a good URL and other numbers for different types of problems. The function will be run on each URL queued for access checking.

Within a profile, links may be used as objects with some useful methods available, such as httpStatus (to check for 404 Not Found or 403 Forbidden), contains (to check if the contents contain a string), and isPDF (to confirm an expected content type).

```php
<?php

require_once "MacProfile.php";

class ScienceDirect extends MacProfile {

    function verify_url( $url ) {
        if( is_string( $url ) ) $url = new URL( $url );
        if( $url->httpCode() != "200" ) return self::BAD_HTTP_CODE;
        if( ! $this->verify_url_access_string( $url ) ) return self::NO_ACCESS;
        if( ! $this->verify_url_has_working_pdf( $url ) ) return self::NO_PDF;
        return self::OK;
    }

    private function verify_url_access_string( $url ) {
        return true;
        return $url->contains("You are entitled to access the full text of this document" );
    }

    private function verify_url_has_working_pdf( $url ) {
        foreach( $url -> getHTMLDocument() -> getLinks() as $link ) {
            if( substr( $link->getUrl(), -9 ) === "-main.pdf" )
                return $link->isPDF();
        }
        return false;
    }
}
```

Implementation details

Normac is open source software written by me in collaboration with my husband Jack Schmidt, and the source code is available from the GitHub repository: https://github.com/zemkat/Normac .

The code is written in PHP/MySQL. MARC records may be submitted as binary MARC or as mnemonic text marc (.mrk).

Within Normac’s code, MARC record representation is object-oriented, making available functions straightforward to read and modify.

The command line version of the normalizer does not require a database and can be run on Unix, Mac OS X, or Windows (tested under Cygwin). The web version has been tested on a Linux / apache server and major web browsers.

The access checker requires a database for link queuing, and currently runs on MySQL under Linux.

Future plans

The current version of the software requires some installation of software either on a server or desktop computer. If this requirement is prohibitive, the normalizer may be rewritten in JavaScript, allowing the process to run completely by visiting a web page, with no need to upload files to a server.

Your contributions?

I have included functions that I have found to be common types of edits so that they may be easily shared and modified. I welcome contributions of functions for new kinds of edits, or suggestions of ones that would be useful.

I have also included platform profiles for detecting access issues for some vendors with which I have encountered them. If you find ebooks whose access is being incorrectly detected, please let me know. I also welcome submissions of new vendor profiles to the repository.

Notes:

[1] PyMARC [Internet]. Available from: http://pymarc.sourceforge.net/

[2] MARC/Perl [Internet]. Available from: http://marcpm.sourceforge.net/

[3] MarcEdit Homepage [Internet]. Available from: http://marcedit.reeset.net/

[4] Provider-Neutral E-Monograph MARC Record Guide [Internet]. Library of Congress. Available from: http://www.loc.gov/aba/pcc/bibco/documents/PN-Guide.pdf

[5] RDA Toolkit: Resource Description & Access [Internet]. Available from: http://www.rdatoolkit.org/

[6] OCLC RDA Policy Statement [Internet]. Available from http://www.oclc.org/rda/new-policy.en.html

[7] W3C Link Checker [Internet]. World Wide Web Consortium. Available from: http://validator.w3.org/checklink

[8] Xenu’s Link Sleuth [Internet]. Available from: http://home.snafu.de/tilman/xenulink.html

[9] HTTP/1.1: Status Code Definitions [Internet]. World Wide Web Consortium. Available from: http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html

About the Author

Kathryn Lybarger (Kathryn.Lybarger@uky.edu) is Head of Cataloging and Metadata at the University of Kentucky Libraries. Her interests in libraries include metadata of all sorts (including traditional cataloging and fancy new standards), preservation, digitization, and automation / computer-assisted workflows.

Subscribe to comments: For this article | For all articles

One Response to "Keeping up with Ebooks: Automated Normalization and Access Checking with Normac"

Please leave a response below, or trackback from your own site.

1. Tim McCarthy, 2015-03-13

   Very interesting article. Since it’s from 2013, I’d love to hear how people are using it and what platform they’re running it on.