E-journals: Can you See me now?

By
John Wynstra
University of Northern Iowa
http://www.uni.edu/~wynstra/iug2003

Presentation Goals

• Who
  • What
    • Why
      • How
        • Where
Getting to Know You
- Academic/Public/Special/School
- ER databases? Why?
- E-journals in Catalog?
- plan to?
- Use Serials Solutions
- Use a vendor other than Serials Solutions
- Looking into Link Server Technology
- Like e-journals, hate e-journals

UNI Rod Library
- University of Northern Iowa
  - 13,000 students
- Rod Library (CVLC)
  - 900,000 Books & Bound Periodical volumes
  - 164 subscription databases
  - 15,000 Electronic Journal access points
  - 60 full time staff (22 library faculty)
  - 100 plus public Workstations
Electronic Resources Committee

- 2 Library Information Technologies
- 2 Reference Department
- 2 Cataloging Department
- 1 Acquisitions Department
- 1 Collection Management Department

Library Info Tech Specialist

- Innovative System Admin (1 of 3)
- Web Application Developer
- Librarian
- My Role in this project
  - Programmer/integrator
What are we Doing?

- Getting an inventory of full text e-journals from Serials Solutions
- Generating brief bib records in Marc format using the 6 data items provided in the Serials Solutions inventory
- Loading them into our Catalog

E-journal Inventory data

- # Titles = 11,000 (14,000 + access points)
- #Vendors = 16
- #Data Elements
  - Title
  - ISSN
  - Start Date
  - End Date
  - Provider
  - URL
- Cost = 2000 plus
Brief Bibs in Marc Format

- 1 record per title (not per access point)
- Added Author for provider
- Coverage in Notes
- Proxifying the URL
- [Electronic Resource] gmd
- Load Table instructions (949)
- Add item record
- Genre index

Updating the catalog

- Load new records (note record ranges)
- Unload old records (using record range)
- Time = 6-15 hours every two months
  - Serials Solutions profiling - 6-10 hours
  - Transforming data – 1-2 hours
  - Loading records - 3 hours
Why?

• Short Answer – Single source of data
• Long Answer - Natural step in development of ER database
• Excuse - Lack of Better Alternatives at the time

Guiding Principles

“Leverage the Catalog”
“Catalog for retrieval and not just for description”

-Jerry Caswell-
The Process in a Nutshell

- Update Serials Solutions profile
- Receive Dataload from SS(.html,.csv)
- Data Checklist
- Remove column headers
- Run SSTransform
- Run Marc Maker (marc edit)
- FTS file up to Catalog
- Preprocess
- Load
- Remove Old Data
- Test

Tools

- Perl Interpreter (Active Perl)
  - http://www.activestate.com/Products/ActivePerl/
- Perl Script – SSTransform.pl
  - http://www.uni.edu/~wynstra/iug2003
- Marc Edit
  - http://oregonstate.edu/~reeset/marcedit/html
- Innovative FTS
How does the program work?

- Open 2 files
  - Input = “rep.csv”
  - Output=“marc_ready.txt”
- Read Title and Title Details (1 line at a time)
  - Proxify URLs
  - Print to “marc_ready.txt”
- Close files – DONE!

Demo

- How does the program work?
  - Open, read, transform, print
- Show Rep.csv
- Show output of SSTransform.pl
- Show Marc Edit in action
- Show SSTransform.pl Code
Customizing the Program

- These print marc_ready statements need to be adjusted for your institution. You may want to adjust others (delete or edit) or add your own.

```plaintext
= LDR = 040
= 001 = 049
= 003 = 599
= 005 = 856
= 008 = 949
```

LDR

```plaintext
print marc_ready
"=LDR 00000nas\\\2200000\a\\4500\n"
```
print marc_ready
"=001 SS-$i\n";

This number is automatically generated within SSTransform. You may want to Change the SS prefix, but you don’t have to.

Example
print marc_ready "=040 \\\\$aN\$cNIU\n";
005 Date and Time of Last Transaction

print marc_ready
"=005 20030211000000.0\n";

You may want to set this every other month when you run the program. I never got around to automating this.

008 Fixed Length Data Elements

print marc_ready
"=008 020401nuuuu\\\\xx\\\\\\\\\\\\\\\\\\en g\\d\n";

Update(mmyydd) the date here. Be careful of the \\.  \ is equivalent to a blank.
040 Cataloging Source

Example

print marc_ready "=040 $aNIIU$cNIU\n";

Replace NIU with your own symbol.

049 Location Code

print marc_ready

"=049 $aNIIUU\n";

Replace NIUU with your own location code.
Print:

```
599

print marc_ready
"\=599 \\\\$/a\nUNI: Full access restricted to individuals affiliated with UNI.
"

Change the red portion to whatever you want or delete the whole line altogether.
```

856 URL

If you want to bypass the proxify subroutine change the following two lines:

```
$proxyurl = proxify ($ur[$s],$prov[$s]);
print marc_ready "\=856 40\$/zConnect to $prov[$s]\$u$proxyurl\n";
```

To this:

```
#$proxyurl = proxify ($ur[$s],$prov[$s]);
print marc_ready "\=856 40\$/zConnect to $prov[$s]\$u $ur[$s]\n";
```
949 Local Load instructions

```plaintext
print marc_ready
"=949  \\\$
a*recs-b;bn-1intr;b3-s;\n"
print marc_ready
"=949  \1\$l1intr\$t15\$se;\n\n"
```

The information in this field is specific to our load tables, location codes, etc. You may reconfigure or delete these two lines.

---

Customizing Proxify

- Subroutine proxify

```plaintext
Comment out the 12 lines begin at $testphrase1
```
Thank You

• Have a nice day
• Good Luck
  – John.wynstra@uni.edu