

November 2007

People Profile: Carol Ou

Editor

Follow this and additional works at: <https://docs.lib.purdue.edu/atg>



Part of the [Library and Information Science Commons](#)

Recommended Citation

Editor (2007) "People Profile: Carol Ou," *Against the Grain*: Vol. 19: Iss. 5, Article 23.

DOI: <https://doi.org/10.7771/2380-176X.5290>

This document has been made available through Purdue e-Pubs, a service of the Purdue University Libraries. Please contact epubs@purdue.edu for additional information.

metasearch application. Maintaining updated lists of database links is arguably just one part of the larger problem of helping users select which of the library's various licensed databases is most appropriate for their search. Federated search and metasearch applications run around this problem by enabling users to search a number of databases all at once instead of individually. Some of these applications also seem to organize, and allow centralized updating of database information, which would help address the original problem of maintaining database links. **Tutt Library** did not have a federated search or metasearch application, however, and we had no immediate plans to implement one, so this possible solution was ruled out.

The advantages of constructing a system to dynamically generate these database pages seemed obvious. It would allow a staff member to update, for example, a changed URL in just one record instead of having to search for that URL and edit it across many individual static pages. And, though this had not been a major issue, database information across the entire Website could not help but become presented in a more consistent manner, since all the links and descriptions would be derived from the same store of information.

As we considered the notion of building a separate, local database to house this database information, there was also the realization that the library was already supporting a mature system designed to house and help maintain similar kinds of information. This was the integrated library system, locally known as **TIGER**, running on **III Millennium** software. Like many libraries, **Tutt Library** was already storing a great deal of access and descriptive information in the ILS, in the form of MARC bibliographic records. The ILS also had familiar client software for adding and editing this metadata. Additionally, we had already cataloged the vast majority of our licensed electronic databases in this ILS that we were then separately listing on the Website. To use the bibliographic records already in **TIGER** as the basis for dynamically generating database pages for the library's Web presence was definitely an intriguing prospect, and given our small staff and limited resources, perhaps even a wiser decision.

To create a separate local database of databases would have reduced the places to update database information from many down to two: the ILS and the new database of databases. To further reduce the list of places to update this information down to one was even more appealing. The decision was made to create dynamic database pages that would be automatically populated with data from MARC records in **TIGER**.

This new process requires that we add more information to the MARC record as a database is cataloged. Some of this information is not "standard" cataloging, but falls into the realm of local practices. We tried to use fields within the MARC record that would not cause prob-

against the grain people profile

Systems Librarian, Colorado College Tutt Library
1021 N. Cascade Ave. Colorado Springs, CO 80903
Phone: (719) 389-6895 Fax: (719) 389-6082
<carol.ou@coloradocollege.edu >

Carol Ou

PROFESSIONAL CAREER AND ACTIVITIES: Currently Systems Librarian at Colorado College Tutt Library.

HOW/WHERE DO I SEE THE INDUSTRY IN FIVE YEARS: Further pulling together of information from various library systems. This project to pull data out of our ILS and expose it via a specific Webpage format is hopefully just a first baby step. 🐛

lems with standards or other elements of our catalog database. Specifically, we made these changes in the records:

1. Edit 85640: |zView online|uhttp://... proxy the 856 (references the proper URL for access to the title)
2. Add 946: ccweb (used for sorting out which records to use to create the list of databases)
3. Add 590 with description supplied by ordering librarian. Add Coverage: at end of Description (provides summary and dates of coverage as an annotation on the Webpages)
4. If title should appear different on database pages from 245, Add 2461 |iColorado College:|aOther title (provides title cross-references)
5. Add 690 with localsubject so title will appear on corresponding subject database pages. (used to generate subject Webpages)

The department previously had one staff member catalog the databases on **OCLC** and add the records to our online system and another person add links to our Webpages. One person now does the cataloging, adding the special fields which are used by scripts to create the Webpages, so we are spared all the work of making the pages separately.

Though we would be making use of the intellectual work contained within the cataloging records to populate our new dynamic database pages, we still had to build mechanisms for effectively transferring and displaying this information. For these tasks, we chiefly made use of **MySQL** for the database component and **PHP** for the necessary programming and to dynamically generate the Webpages. **MySQL** and **PHP** are both Open Source tools readily available on many Web servers, including the one used for the library's Website.

Though the bibliographic records for the databases would be stored in the database of the ILS, it was determined that repeatedly hitting our catalog server to look up and read those individual records would be unwise. Instead we opted to select all the ccweb database records

using **Millennium's Create List** function and export specific pieces of their bibliographic records in a tab-delimited text file, to be imported into a separate **MySQL** database. The dynamic database pages would actually be driven by data stored in this **MySQL** database, instead of directly from data stored in **TIGER**.

The necessary **MySQL** tables were relatively straightforward, with one table containing the bulk of the database records (including descriptive information and the URLs), another table indexing the multiple titles (main and alternate) associated with each record, and another table indexing the multiple local subject category assignments that might be assigned to each record.

The necessary **PHP** programming fell into two groups. There was one set of scripts designed to take the tab-delimited export from **TIGER**, parse the various MARC fields and import the results into the **MySQL** tables. There was another set of scripts designed to take the data in the **MySQL** tables and generate Webpages with the same look and feel and intent as the existing database pages. That is to say, this second set of **PHP** scripts needed to be able to display database information in the same existing ways: as a standard A-Z list where all of the links were proxied for off-campus access, as individual subject pages listing only the databases assigned to a particular local subject heading, and as an A-Z list where all of the links were direct to the database and none of the links were proxied. Because of all the work done in cataloging to organize the data fields, the programming to import the data and then display it turned out to be very straightforward.

The new database pages have only been live for a few weeks at the time of this writing, but there were a few lessons learned in the ramp up to implementation. The new database subject pages were not organized identically to the previous database subject pages. Namely, the new pages list their databases in alphabetical order, whereas in some of the previous database subject pages, the liaison librarians had manually organized databases in a different fashion.

continued on page 56