

SCHOLARLY PUBLISHING OFFICE WHITEPAPER\*

## Plan for implementing a web-based unified bibliographic database for SPO publications

Kevin S. Hawkins

### Background

Currently SPO uses FileMaker databases for inputting and generating the header portion of Text Class documents and bib records for Bib Class collections. SPO maintains separate FileMaker databases for each DLXS collection: each has its own interface and field(s) with *field calculations* that result in one or more chunks of XML-like text used for various parts of the prepping process.

### Motivations

There are a number of disadvantages to the current system of having separate FileMaker databases for each publication:

- There is no way to enforce standard encoding across collections.
- The use of field calculations predates XML support in FileMaker, so the output is only checked for validity when running a prep script. (A more XML-centric workflow would have SPO staff export FileMaker databases as XML and then transform them with XSL.)
- Access to these databases can only be provided by DSS hosting them. Security issues make remote access to both version 6 and 7 databases difficult, so Panagis would prefer that we not use FileMaker at all.
- Inputting non-ASCII characters presents problems: FileMaker won't store a character for which a glyph is not available in the font being used in the FileMaker form. This will continue to be a problem as long as there are no fonts with the same name available for Windows and Mac OS X that have all non-ASCII glyphs needed. (A lack of appropriate fonts available for Windows and Mac OS X would also be a problem in any other solution, but at least most programs and web interfaces are capable of storing a character even if it can't display it.)

### Proposed unified bibliographic database

SPO needs a unified database for all Text Class headers and Bib Class records that would store the data uniformly and have one export mapping for each of (a) Text Class headers, (b) Level-2 DIV-BIBLs, and (c) Bib Class records. In addition, we need (d) the capability to set up custom mappings, as for the article list in the Diderot project.

These mappings need to be:

- generic to work for all collections
- close to DLPS practice for consistency across the DLXS repository
- close to TEI practice for ease of future migration and reuse of content

---

\* This work is licensed under a Creative Commons Attribution 3.0 License. To request permission to use this content in a way not allowed by the Creative Commons license, contact [copyright@umich.edu](mailto:copyright@umich.edu). © The Regents of the University of Michigan, 2007.

The database and interface should be hosted on servers maintained by Core Services for maximum integration with other SPO systems and for flexible server access for development.

There should be a web-based interface for editing the database with differentiated access: SPO staff should have access to the entire database, and content providers should have access only to the records for their collection(s).

### **Steps to implementation**

1. Develop a database scheme that allows for all types of content found in Text Class headers and Bib Class records.
2. Develop export mappings of database fields to XML elements for (a) through (d) above.
3. Change all "Export" fields in existing FileMaker databases to match the output generated by the new export mappings, and reindex affected collections using this modified export data, closely verifying that filtering and region searching are unaffected. Note changes so that Jeremy can adjust collection-specific XSL files for generating bib files.
4. Create the new database and develop the corresponding web interface.
5. Coordinate with all students, remote content providers, Kevin, Shana, and possibly Terri to halt updating of FileMaker databases being absorbed into new database.
6. Export data from the existing FileMaker databases and import into the new database.
7. Redirect everyone to use the new database for now on.

### **Future directions**

With a unified bibliographic database, SPO can easily make global changes to all collections, to be reflected as each collection is updated. For example, copyright status, Creative Commons licenses, or other metadata might be added to records at a later time.

### **Outstanding issues**

We need to decide whether to host the database and interface on development or production servers.

We also need to determine whether authorization to the web interface should use the Library auth system or campus LDAP system.