Cobre: A Comparative Book Reader for Los Primeros Libros

Benjamin Liles, James Creel, Alexey Maslov, Steven Nuernberg, Anton duPlessis, Holly Mercer, Mark McFarland, John Leggett

Texas A&M University Libraries
5000 TAMU
College Station, TX USA
+1 (979) 845-8111
{bliles, jcreel, ak_maslov, snuernberg, duplessis, hmercer, leggett}@library.tamu.edu

The University of Texas Libraries
1 University Station S5400
Austin, TX USA
+1 (512) 495-4129
mcfarland@Austin.utexas.edu

Abstract

We describe the design, implementation, and evaluation of a comparative book reader that supports the scholarly activities deemed most important by cultural history scholars. A filmstrip metaphor allows rapid access to various components of the books. Parallel filmstrips allow comparison among multiple copies of the same title to fill in for missing pages or other maladies, and to study page-level annotations to help in interpreting the content. Playing the filmstrips allow the scholar to search for or discover items of interest by watching the pages of the books flow past. The need for a canonical book and possibly multiple “Frankenbooks” is described and the implementation is given. The architecture of the tool and the programming methodology are also described. The results of a discount usability study are presented along with lessons learned and future directions for improving the comparative book reader.

1. Introduction

The Los Primeros Libros (LPL) project is building a digital collection of the first books printed in Mexico before 1601. These monographs are very important because they represent the first printing in the New World and provide primary sources for scholarly studies focused on cultural history. Approximately 130 known unique titles of approximately 220 produced are held in institutions around the world. While it is a specific goal to acquire at least one example of each unique surviving imprint, it is important to acquire as many copies as possible, since many of these books are damaged in various ways and marginalia, typographical variants, ownership marks, and other copy-specific attributes are often critical for interpretation and other purposes.
conditions of the books, as well as user stories [12] from cultural history scholars associated with the project. The books range in age from approximately 410 to 460 years old and have maladies such as: missing pages, page ranges, covers, and bindings; worm holes; scribbles obscuring text; misbound rebindings; severe warping; fire and water damage; and worn pages due to extreme use. Working with scholars from the Asociación Mexicana de Bibliotecas e Instituciones con Fondos Antiguos (AMBIFA) [3] over the last five years, we have been able to identify user stories that describe the desired scholarly interactions with these books. These user stories create requirements for comparative book readers that fall into three main areas: content, structure, and abstraction.

2.1. Content

The user stories in this area have to do with reading and understanding the textual content and marginalia.

2.1.1. User Story 1. Users want to locate the books and basic information about the books, printers, authors, funders, etc. Satisfying this user story requires appropriate metadata and indexes over the metadata for searching and browsing. The normal tradeoff exists between the number of metadata fields and the actual capture of quality metadata for those fields.

2.1.2. User Story 2. Users want to read, browse, and study the books. Satisfying this user story requires user interfaces to the books that allow for reading, page turning, navigating through the structure, and zooming in/out for studying details. Several book readers have been developed that allow these kinds of operations [10,40,51].

2.1.3. User Story 3. Users want to read/study marginalia, drawings, and printer’s text, within and across all copies of the books to increase understanding of certain passages. Satisfying this user story requires user interfaces to the books that allow zooming in/out for studying details, synchronous page viewing from multiple books, and the capability to align one book with another.

2.1.4. User Story 4. Scholars want to use these books as teaching and learning materials for religious studies, history, cultural history, linguistics, and language throughout the world for free. Satisfying this user story requires the user interfaces discussed above, open access to all books and supporting materials, and a widely accessible format that students can use on their personal mobile devices.

2.2. Structure

The user stories in this area have to do with locating graphical components and physical structures of the books.

2.2.1. User Story 5. Users want to easily locate and inspect various components of the books such as approbations, portadas, marcas de fuego, engravings, and capitulas. Satisfying this user story requires user interfaces to the books that allow rapid browsing and zooming in/out for studying the details.

2.2.2. User Story 6. Users want to quickly identify individual pages, as well as out of order or misbound pages. Satisfying this user story requires a user interface to the books that provides the capability to align a book to a standard. These books are sometimes rebound individually and sometimes rebound with other books. As an example, scholars at Biblioteca Lafragua discovered, through careful study and much effort, four missing pages from a Bulla [46] that were misbound in a compendium of several books [38].

2.2.3. User Story 7. Users want to refer to all copies of a book to fill in for missing or unreadable text due to warping, worm holes or missing (parts of) pages. Satisfying this user story requires user interfaces to the books that allow synchronous page viewing from multiple books and the capability to align one book with another.

2.2.4. User Story 8. Upon discovering one of these books in their physical collection, a user wants to quickly find out how complete their copy is and what might be missing. Satisfying this user story requires a user interface to the books that allows the capability to align a book to a standard.

2.3. Abstraction

The requirements in this area are derived from the user stories above.

2.3.1. On the need for a Canonical Book. User Stories 3, 6, 7 and 8 require either aligning one book with another or to a standard. Since many of these books are missing pages in arbitrary locations, it is very difficult to align one against the other, and in our
case we need to align an arbitrary number of books. A canonical book (abstract but complete structure with abstract pages) allows each individual book to be aligned with it separately. An arbitrary number of books can then easily be aligned for comparison.

2.3.2. On the need for Frankenbooks. In addition to User Stories 3, 6, 7 and 8 which argue for a canonical book, User Stores 4, 6, 7 and 8 make the case for Frankenbooks. A Frankenbook (with a nod to Shelly [52]) is a canonical book that has actual page images, drawn as desired from any existing copy of a book, replacing the abstract pages of the canonical book. A Frankenbook makes alignment and recognition easier for User Stories 6 and 7 by having high quality pages in a single copy. While there is generally a need for only one canonical book, there may be several Frankenbooks per title. We have found different full-page engravings in different copies of a Bulla [46] and one can only imagine the variants scholars will discover when studying these books via Cobre. Multiple Frankenbooks allow one to satisfy the user story requirements above and other specialized needs with the best possible reference book.

One question that arises is: When is a book a copy (with variation) of another book and when is it an edition of the book? Bibliographers generally agree that if fifty percent or more of a book has been re-typeset, then the book should be considered a new edition [20]. In our case, we do have books that have been re-typeset. Type was a precious commodity and one could not afford to store type for future printings. This would argue for multiple canonical books, one for each edition. But scholars want to satisfy User Stories 1-8 across all copies of a book. This argues for alignment of all copies to one canonical book. At the current time, we generate one canonical book and align all copies to this standard. This requires that we add “pages” containing the phrase “for synchronization only” in the appropriate places.

3. Cobre

Cobre [13] was iteratively designed, implemented, and evaluated (see sections 3.2.3. and 5. below) based upon the requirements derived from the user stories above. User story 1 is satisfied by the metadata standards [44] adopted by LPL project participants and the searching and browsing capabilities of the main project website [32] and repository software [53]. Cobre satisfies the remaining user stories, as discussed below.

3.1. Reading and Editing Tools

Cobre is a web application that offers a variety of specialized tools facilitating interaction with the Los Primeros Libros collection. The reading tools are available to all visitors and support both casual reading and scholarly interaction with the books. The editing tools are available only to authorized users and are used to create and modify the book structures.


**Book Overview.** The book overview (Figure 1) provides all of the available metadata describing the book along with icons providing navigation to the reading view, detailed view, and repository view. Icons are also available for downloading a PDF copy and adding the book to a list for the comparison view.

**Reading View.** The reading view (Figure 2) starts with the front cover and progresses through the interior of the book, two pages at a time, to replicate the behavior that is available when reading a physical book. Users can either use the arrow keys on their keyboard or click on the pages to navigate through the book. Additionally, each page has an icon linking to the detailed view of that page.

**Figure 1. Book Overview.**
Detailed View. The detailed view (Figure 3) provides a zoomable interface for each page and a filmstrip view of a run of pages. The page can be dragged around the image viewer pane and zoomed using mouse clicks or a scroll wheel. Movement and zoom controls are overlaid on the viewer. The overlaid open book icon links back to the reading view of the book opened to the current page, while the printer icon links to a high-resolution image that is suitable for printing purposes. A scrollable and playable filmstrip of the pages is displayed beneath the detailed image viewer. The filmstrip centers the current page being viewed and provides context before and after this page. Clicking on a thumbnail in the filmstrip will load that page in the image viewer pane. The filmstrip provides three thumbnail sizes and can be played at six different speeds.

Repository View. The books are hosted in a DSpace [53] repository. The repository view (Figure 4) is a custom DSpace/Manakin [45] theme that displays tiled page thumbnails. Each tile contains links to the other views available in Cobre, as well as a quick link to a larger preview of the page.

Comparison View. The comparison view (Figure 5) displays parallel, scrollable and playable filmstrips of selected books and is at the heart of satisfying User Stories 3, 6, 7 and 8. As in the detailed view, the filmstrips provide three thumbnail sizes and can be played at six different speeds. This allows for quick and easy comparison of page images from multiple books.

The browse, search and view interfaces provide methods to add books to a comparison list. When two or more books have been added to the list, the comparison view is accessible by clicking on the “compare” button. Locking the filmstrips together and scrolling or playing through the pages allows a quick comparison of the pages of the books. Different copies of the same book that have been individually scrolled can easily be synchronized with a particular filmstrip (book) by selecting its sync icon. Individual pages from each book being compared can be inspected at the same time in a collateral quick zoom view (Figure 6).
3.1.2. Editing Tools. Various features of the books can be edited by authorized users (curators, scholars, etc.) by using the integrated editing tools. The basic page editor allows for the reordering of pages as well as adding and removing pages as necessary. Augmentation of the basic page editor adds the functionality to create Frankenbooks and to align copies of a book to the canonical version. The annotation editor provides for adding, editing, and removing annotations or copying the annotations from one copy of a book to another. The annotations discussed here are descriptions of structure, special markings, etc., as one may find in a critical bibliography [20].

Basic Page Editor. The basic page editor (Figure 7) allows for adding, removing and reordering pages using a drag and drop interface. Any page can be dragged from its current location to any other location in the filmstrip. Pages are added by dragging a new, blank page and dropping it in the desired location in the filmstrip. Pages are removed by dragging them from the filmstrip and dropping them on the delete page icon.

The book that is being edited can be compared with another book. Comparing books while editing is useful for aligning books with the canonical version (Figure 7). To aid alignment, pages of the canonical version display the annotations on the page when the page is clicked. The user adds blank pages into the book that is being edited as necessary to account for pages that are missing with respect to the canonical version.

Frankenbook Editor. The basic page editor functionality is extended when editing a Frankenbook. In addition to the standard operations, pages from the compared books can be dragged and dropped onto pages in the Frankenbook (Figure 8). When there are multiple copies of a book, the user can select which copy has the most complete version of a page, on a page by page basis, or in the case of unique attributes, which copy of a page has the most desirable attributes (for example, the most interesting marginalia).
Annotation Editor. An annotation editor (Figure 9) is also available that allows for adding, editing, and removing annotations on existing copies, canonical copies, and Frankenbooks. The user inputs the starting page and length of the annotation, whether the annotation is part of a structural hierarchy, and text of the annotation itself. The annotations can be input in the context of an existing copy and then duplicated in the abstract canonical copy, or may be input directly in the context of the canonical copy if they are available from some external source, such as a critical bibliography [20,55]. Structural annotations provide support for describing sections and chapters in books. Non-structural annotations are used for annotating special markings, marginalia and other items of interest.

3.2. Architecture

Open access and preservation of the data used in Cobre are part of the broader goal of the Los Primeros Libros project [44]. To that end, all of the data (page images, metadata, and generated structures) are stored in DSpace, an open access, institutional repository, using accessible formats [53]. The repository itself is part of a broader effort by Texas institutions to ensure preservation of scholarly materials [56].

In the DSpace data model, individual copies of books are represented by Items. Descriptive metadata about a book is stored with the Item as a Dublin Core derivative. Page images, derivative image versions, and supporting files are stored as Bitstreams. Structural metadata (page order, annotations, etc) is encoded in XML and stored as a Bitstream as described in section 3.2.2.

Although DSpace/Manakin [45] provides extensive customization options, the difficulty of rapidly developing complex user interfaces necessitated the creation of a separate web application. Django [15] was selected as the framework for the web application based on the desired features of Cobre.

In Django, Models are used to represent books, pages and annotations. Model attributes on the Book Model are used to encode the descriptive metadata and the canonical version relationship. The Page Model consists of references to the Bitstreams stored in DSpace as opposed to copies of the images.

3.2.1. Integration. The DSpace repository is the primary data store for Cobre while the Django application is the primary user interface. To support this duality, descriptive and structural metadata about the books are harvested from the repository using the repository’s OAI-PMH [42] and OAI/ORE [41] interfaces (Figure 10). The book metadata is then transformed into the data structures used in Django.

The Django reading and editing tools interact only with structural metadata and references to the Bitstreams stored in the repository. When editing a book’s structure in Django, all of the modifications are stored in the Django application which also provides a method for extracting the modified structural metadata as a single file that can then be uploaded (as another Bitstream associated with this Item) into the repository for permanent storage and preservation.

3.2.2. Structural Metadata. In the context of Cobre, a book’s structural metadata encapsulates the following concepts: ordering of pages (existing and placeholder), annotations, canonical reference, and other page metadata. Existing pages contain references to the Bitstreams stored in the repository [53]. Placeholder pages indicate that a page is missing or that the page is needed for synchronization with respect to the canonical version.
In the repository, the structural metadata is encoded as XML that conforms to a simple XML schema. The schema defines book attributes for indicating if a book is canonical or has a canonical version. The top level elements of the book are structure (which contains annotation information), exterior (which contains page references for external views of the book), and interior (which contains page references for the internal pages of the book). The interior page references are grouped into chunks that can represent a single page or a collection of placeholder pages. This structural metadata schema is used to describe all of the books including existing, canonical and Frankenbooks.

3.2.3. Technologies Used. JIRA’s GreenHopper Agile software development environment [21] was used to track and manage the development process based upon the user stories discussed in section 2. The user interfaces and system integrations are enabled by a variety of open source libraries: Djatoka [16], a modified OpenLayers map viewer [16], and jQuery and its extensions [24,25,26]. In addition, we developed a new extension to jQuery to add specific filmstrip visualization features for the Detailed and Comparison views (section 3.1.1).

Foundational storage and hosting for the books is provided by the DSpace [53] repository platform. The user interfaces at the repository level are implemented in the Manakin XML UI [45]. The remaining user interfaces are implemented using the Django [15] web application framework. The OAI-PMH [42] interface of the DSpace repository is utilized along with OAI-ORE [34,41] aggregations to enable harvesting of the books by Django. The Django harvesting is enabled by the Python [48] OAI-PMH library, pyoai [47], and a DSpace specific Python interface [49]. The search interface in Django is enabled by combining the Haystack [22] search integrator with the Apache SOLR [2] search server.

4. Related work

Book readers are not a novel concept (for a history and timeline of book readers going back to 1971 see [6]). Early digitization efforts motivated tools that would enable users to effectively find and utilize digital documents. Over time, software and hardware designs were refined to satisfy some of the requirements outlined in section 2. Today, basic reader functionality is available on dedicated physical devices (Amazon Kindle, etc.), as software for personal computers and mobile devices (Apple iPad, etc.), and as web interfaces (Google Books, etc.). Many digital libraries also offer specialized reader views, including comparative views, of books in their collections, either as primary access points or experimental projects [29,43,58]. Research continues on augmenting the reading experience through improvements in display quality, browsing support, visualizations, and annotation systems [8,10,30,31,33].

Most efforts described above strive to improve the reading experience and simplify retrieving specific information. The Cobre reader, on the other hand, supports reader interaction and scholarly work in some novel ways, as defined by scholars in multiple academic fields [3,14,20,23,55]. Cobre supports comparison tools that facilitate: synchronous examination of multiple books, parallel comparison, collation of pages from different sources, creation of a single master copy (canonical book), and creation of multiple different reference copies (Frankenbooks). Additionally, Cobre supports the implementation of a simple notational structure flexible enough to accommodate readily identifiable aspects of a book, such as a table of contents, as well as the nuanced markings of specific pages of inquiry, such as engravings or firemarks [4]. The design of the interface itself leverages several user interface design techniques such as animated scrolling [28] in the filmstrip views and tiled thumbnails with quick previews in the repository view [11].

5. Evaluation

Unlike many prototypes that are designed, implemented and then tested by the intended user community, Cobre has benefited from a user-driven and iterative design, implementation and testing approach. The desired functionality and user interfaces were tested and modified (if necessary) every few weeks at the Sprint reviews required by the Agile/Scrum/User Stories [12] methodology. This allowed Cobre to be evaluated many more times (essentially continuously) as the collection was (and still is) being built.

In addition, to evaluate the user interface and repository aspects of Cobre, we employed a heuristic evaluation discount usability technique as described in Kane [27] and Nielsen and Molich [39]. We provided documentation on heuristic evaluation and the use of discount usability with Agile methods [54] to a repository content expert and two web application and user interface experts. We asked them to use Cobre for two hours to accomplish the tasks for which Cobre was designed as described in User Stories 2 through 8. Neither person had previous experience with Cobre and we purposefully did not provide any
documentation or training. They simply reported their experiences, both positive and negative.

Overall our expert evaluators found that the user interface was intuitive, the controls were consistent, and they were able to easily accomplish User Stories 2 through 8. Example comments included: “I easily was able to access a book and understood the difference between the reading view and the detailed view”; “The icons … used familiar conventions, and the hover text is clear”; “Zooming in/out was easy and I was able to view details quite well”; “I was able to select multiple books for comparison, and could align the selected books”; “As someone unfamiliar with the tool and with the content, I was able to navigate, explore and use the interfaces freely”; “It is easy enough for a novice to use, but the features seemed powerful enough for more intensive scholarly use”; “…with the canonical form of a given work, I was able to locate and view various components/structural markers in different copies of the works”; “The system certainly addresses real world needs (e.g., close study and comparison of rare books, open access)”; and, “The icons and hover text generally provide enough help, and it is certainly concise and task-centered.”

As expected with expert evaluators and heuristic evaluation, our evaluators were able to identify some inconsistencies/oversights and opportunities for improvement of Cobre. Example comments included: Cobre “uses a standard convention to signal an image is loading in the reading view, but there is nothing to indicate an image is loading in the detailed view”; “The lock icon should have hover text that explains what it does – all the other icons have this, and it is just the right amount of guidance”; when moving from reading view to detailed view “…it was difficult for me to tell what page in the sequence I was viewing … with no marker to indicate ‘This is the page currently being viewed’ there is no context for more text-heavy pages and I found myself getting lost”; “…the icon for the detailed view on the landing page for the ‘Start Reading’ link was different from the icon for the detailed view on the metadata record”; in the filmstrip view, I “wanted to click and hold on the navigator” controls “but had to click repeatedly to slide the page into position”; and, “the controls for the filmstrip view were not consistent in the detailed and comparison views.” These deficiencies were straightforward to remedy and have all been fixed.

As a practical example of the success of Cobre for its intended purpose, we have been informed that one of the books in the collection, the 1571 book on Nahuatl by Molina [37], is currently being used as the primary text in a language class at Biblioteca Lafragua. The students had been using low quality photocopies of photocopies, but now can download a high resolution copy to their computers or mobile devices. They are using the book readers associated with the LPL project to study the book and to print a high quality version. Even with all of the work that has gone into Cobre to this point, it should still be considered a prototype for the desired scholarly reading tool of the future for Los Primeros Libros and other like projects.

6. Lessons learned & future work

Many lessons learned from the overall LPL project were translated into actions in the design, implementation and evaluation of Cobre. LPL is a very large collaborative effort among many institutions in several countries. This demands close attention to communication, management, and social issues. We found that the Agile/Scrum/User Stories methodology [12] provided a focus on appropriate-sized programming efforts. The methodology contributed to good communication with our user community of cultural history scholars and allowed us to engage them as early as possible.

On a more practical note, we learned that Cobre was also a good quality control tool. After scanning a book and placing it in the repository, one can simply play the filmstrip to find pages that have been scanned incorrectly. Using a repository for holding books as collections of pages, eases the management of image files with no penalty in performance and with benefits such as scalability and preservation. We also agree with Kane [27] that using a discount usability technique in combination with an Agile programming methodology is a powerful combination since both are highly focused on short time frames and high impact.

Many avenues exist for future work on Cobre. One that we have already mentioned (section 2.3) has to do with graceful integration of editions; another has to do with the sharing of Frankenbooks. Currently, our implementation of a Frankenbook consists of links to appropriate pages in multiple books (collections of images) within our repository. This makes sense for Frankenbooks that will be accessed via our website. But, sharing a Frankenbook as a book is quite different. We will have to copy the pages into a new collection and, more than likely, create a pdf version. This brings up issues of provenance unless we always consider Frankenbooks to be ephemeral.

Obviously, there is much that could be done by adding image and spectral analysis to Cobre to help in alignment and feature discovery. We would also like to apply Cobre to other digital image collections such as natural history and manuscript collections. A strong comparative component exists in both research and
teaching applications with these types of images and we believe this would lead to further improvements to Cobre.

Although Cobre is currently live at Texas A&M University Libraries, there still remains work to move Cobre into production on the main LPL website. We expect to receive requests from our user community for new functionality on a continuing basis after Cobre is in full production.

7. Acknowledgments

The authors would like to thank the LPL project participants and members of AMBIFA for providing understanding of the needs of cultural history scholars. Special thanks must go to Manuel de Santiago and Iván Pérez Pineda of Biblioteca Lafragua in this regard. We would also like to thank Adam Mikeal for initial Django template design and Bekki Campise for her work in arranging the components of this paper.

8. References

[22] Haystack http://haystacksearch.org/
[25] jQuery.scrollTo
[26] jQuery.ajax
http://flesler.blogspot.com/2008/02/jqueryserialscroll.html
[34] Maslov, A., Mikeal, A., Phillips, S., Leggett, J., & Mark, M. “Adding OAI-ORE support to repository


[40] Northwestern University Library’s Book Viewer. http://digital.library.northwestern.edu/books/about.html


[44] Partnership Agreement for Impresos Mexicanos del Siglo XVI. http://primeroslibros.org/PrimerosLibros_Agreement.pdf


[47] Pyoai http://pypi.python.org/pypi/pyoai

[48] Python http://python.org/

[49] Python dspace http://pypi.python.org/pypi/dspace


