

Kelli Bogan (bogan@simmons.edu)
Amanda Potter (pottera@simmons.edu)
LIS 531F: XML
5/2/2007

Creating archiveThing: An Evaluation of Progress So Far

At its most basic level, archiveThing is LibraryThing for archives. However, instead of being a place to share book catalogs, archiveThing shares digital collections and finding aids. This paper will look at the creation of archiveThing. It will begin with an examination of why archiveThing is needed and how it is different from other archival Web 2.0 projects already in existence. It will then discuss the envisioned final product, and the questions and obstacles associated with creating archiveThing. Finally, the paper will analyze what has been done so far and what the next steps are in development.

Why archiveThing?

Web 2.0, a term coined by Tim O'Reilly and John Battelle at a 2004 conference, refers to both the technologies being used and a set of principles being expounded. One of the main concepts associated with Web 2.0 and Library 2.0 is that users should be able to make the web their own through “the technologies often used in connection with Web 2.0...[including] Ajax, blogs, APIs, clouds, CSS, RSS, social networking, tagging, and wikis.”¹ The goal of archiveThing is to create a social network system using XML (eXtensible Markup Language) and the principles and technologies of Web 2.0. archiveThing will allow the users of the archive to be the creators of the site, while also educating them about “traditional” archives. It will also provide the opportunity for

¹ <http://www.infotoday.com/online/may06/OnTheNet.shtml>

informal collections and formal archives to meet in a single location, allowing users to connect and interact over common interests.

In addition to a user-centric framework, archiveThing is also trying to educate archivists and create a new tool for their use. Archivists are not known for taking advantage of the Web 2.0 technologies and other online resources that could serve their community and mission. archiveThing hopes to show archivists how they can use technology to promote their archive and potentially draw in users who may not have realized that such an archival resource existed.

Ultimately, archiveThing is about direct user input, outreach, and education. However, there exist archival websites concerned with these issues, some of which are already incorporating Web 2.0 technologies. In order to understand how archiveThing will take Web 2.0 someplace new for archives, these sites need to be examined in more detail.

One of the most visible examples of Web 2.0 being used by archives is found at the University of Michigan where Elizabeth Yakel, the Bentley Library staff, and a group of graduate students have created the Polar Bear Expedition Digital Collections. The project was created in an attempt to “expand the capability of EAD, make the archival and research experience collaborative and participatory, and challenge the traditional finding aid structure.”² Development of the collection began before many of the Web 2.0 technologies (for example, AJAX) were available so they were unable to incorporate as many Web 2.0 features as they would be able to include now. The primary Web 2.0 technology used is social navigation through link paths and comments. These pathways

² http://polarbears.si.umich.edu/index.pl?node_id=272&lastnode_id=1163

provide a way for the user to recognize new connections that may not have been seen in the traditional hierarchical arrangement of archival finding aids and allow users to interact with each other as well as members of the archive staff.

Another example of a website using Web 2.0 is the BBC's WW2 People's War, which spent three years collecting the stories and photographs of the World War II generation. The result was an archive with over 47,000 stories and 15,000 photographs that was created by the British people, "turning the audience into the content generators and giving them a platform to tell their stories."³ This website makes history come alive for its users, the descendants of the storytellers. More importantly, it is going to become a permanent part of the British Library's United Kingdom Web Archiving Consortium (UKWAC). Thus, this is an example of an informal archive (to some degree), created by and for the people, becoming a permanent part of a formal archive.

There are two American based websites that use similar features and concepts as the BBC's site: the September 11th Digital Archive⁴ and the Hurricane Digital Memory Bank,⁵ which documents Hurricane Katrina. Both of these sites collected stories and images from the public that document these two events. One of the most interesting Web 2.0 features on both sites is the overlay of a select group of images and stories on an interactive map.

So, why create archiveThing when there are obviously many Web 2.0 websites already out there? First, it is a rare example of a website that from its origination combines formal and informal archives. Also, while all of these sites provide valuable

³ <http://www.bbc.co.uk/ww2peopleswar/>

⁴ <http://911digitalarchive.org/>

⁵ <http://www.hurricanearchive.org/>

features, archiveThing is the only site that is driven entirely by user content. The formal archive site examples above do not allow users to add their own content. Instead, they can only comment on the content that is put up by the archive. The informal archives allow users to add content, but they only collect a specific type of content, the materials that fit into their collection policies. For example, the September 11th Digital Archive only collects materials related to September 11th. archiveThing has an open collection policy; the users decide what they want to put up on the site with the only restrictions being on copyrighted materials and materials of a pornographic nature. This type of site is advantageous because it means that users don't have to seek out a site that fits their interests; they can create it to meet these interests and, if they have already placed an appropriate collection on another site (for example, Flickr), archiveThing provides a way for users to direct other archiveThing members to this external site. Now that the need for archiveThing has been established, it is time to turn to an examination of the product itself.

The Vision

When archiveThing is complete, users should be able to make and manage an account, create and maintain collections, add favorites, and tag other user's collections. There will be a community page that allows users to talk about specific collections, interests, and archives in general. The site will also be fully searchable and browseable. In addition, archiveThing will be part of the open archives initiative (OAI), which is a framework that uses the OAI protocol in order to allow archives to be collected, searched, and displayed in a single location.

In creating archiveThing, there were several issues were considered. First, since creating a website is a large task, the project needed to be divided into workable increments. It was split into two phases, with the Phase I focused on creating the visible framework for archiveThing and Phase II devoted to the functionality and special features of the website. In deciding what went into each phase, it became a question of what was doable in the timeframe allotted as well as technical skill levels. Since neither of us comes from a programming background, these elements were immediately placed in Phase II. The search function was originally a Phase I goal but, after conducting research, this feature was too difficult to implement without some of the elements already allotted to Phase II (for example, databases).

Choices also needed to be made about the markup languages and related tools being used. It had already been decided that the framework would be a combination of XML, XSLT (eXtensible Stylesheet Language Transformation), CSS (Cascading Style Sheets), and XHTML (eXtensible Hypertext Markup Language). XML was chosen because of its flexibility and ability to make a self-describing schema, both of which will make changes with this markup language easier.

In addition, it needed to be decided whether any additional markup languages would be included. Since archiveThing is meant to be OAI compliant and is geared towards both users and archivists, it was important to include markup languages that are already being used in the archival field. First, unqualified Dublin Core (DC) was incorporated into archiveThing since it is the namespace used by OAI. Second, Encoded Archival Description (EAD), a markup language developed by the Society of American Archivists (SAA) and the Library of Congress, was incorporated because it is an archival

standard and many institutions are already using it. Since we were creating a framework, it was important that both of these standards be incorporated in Phase I so that Phase II could focus on functionality.

Phase I was set with a completion date of April 25, 2007 and Phase II was to be completed by a date to be named later.

Phase I: XML and Site Design

Phase I of archiveThing has been completed and the current mock-up is available at <http://www.archivething.com> and should only be viewed in Firefox 2.0 or Safari.

During Phase I, the framework of archiveThing was developed, including a site-specific namespace and schema, model collections and profiles, a simple browse function, and information pages. In addition, two simple forms were produced to show what information would be required in order to create a new user or collection. Finally, a site design was developed for archiveThing. These framework components will be examined in more detail below.

The namespace, “art,” was developed in addition to the use of DC and EAD standards to allow control over user profiles and collections. The “art” schema and namespace supplies the administrative and display information not provided by the DC or EAD namespaces. Currently, the “art” schema also contains specialized tags for listing collections in the profile. In the Phase II, these tags may be removed with the implementation of a server-side database.

XML and XSLT were used to create profiles and collections in order to provide a consistent appearance throughout the site and to ensure that the correct information is

displayed. The profile models only use the “art” namespace since these pages will not be harvested by OAI.⁶ The Personal Inventory and Descriptive Splash Page collections use “art” and DC, making them harvestable without delving into the unnecessary complexity of EAD. The Institutional Finding Aids use EAD since it is the archival standard and already in use in other digital archives. While appropriate because of its wide use by archives, writing XSLT for EAD proved difficult due to its repetitive tags. The general information and browse pages and forms have been written under the strict XHTML DTD.

Site design was completed using CSS on top of XHTML and XSLT. The colors were selected to match archival document boxes and folders. Tabs themselves were selected as a familiar and elegant interface.⁷ The code for the tabs was obtained from www.dynamicdrive.com⁸ and modified to match the site.

The browse function was included as a way to access the model collections since a working search was not completed in Phase I. The tree format is effective for small numbers of collections and familiar to most computer users, but will likely become too complex when there are many collections to display. It was created using a combination of CSS and JavaScript from www.javascripttoolbox.com.⁹ Overall, the goal was to create something attractive and easy to use.

Evaluation of Phase I

⁶ <http://www.openarchives.org/>

⁷ Krug, S. (2006) *Don't Make Me Think: a common sense approach to web usability*. Berkeley, CA: New Riders Publishing. p. 79-80.

⁸ Specifically <http://www.dynamicdrive.com/style/csslibrary/item/overlapping-horizontal-tabs/>

⁹ Specifically <http://javascripttoolbox.com/lib/mktree/>

Based on the criteria set out in the archiveThing white paper, Phase I has been successfully completed with some qualifiers¹⁰. This phase was essentially about completing the framework of archiveThing that could be created using XML. While some adjustments may need to be made as we move into the technical parts of Phase II, the XML and its related parts that we have completed so far should serve the site well.

Unresolved issues from Phase I include the partially completed forms and the website's appearance on browsers. The final version of the forms will include JavaScript to reveal or hide sections depending on what the user selected. However, the current XHTML forms provide a good starting point.

Due to time constraints and differences in computer access, the site was designed using Firefox 2.0 so both Mac and Windows users would be able to view it in at least one browser. Versions of Internet Explorer do not display the page properly and do not support the JavaScript enabled trees. It will be necessary to optimize the site so that it functions properly on the most recent versions of Firefox, Safari, and Internet Explorer. Since the site does not display attractively on an 800x600 screen, the screen size of a "typical user," resizing issues will also need to be addressed.

Overall, Phase I provided visualization of the appearance and functions for archiveThing, while allowing us to learn about building sites with XML, XSLT, XHTML, and CSS, to use namespaces, and to integrate current XML standards into a new site that will allow it to be interoperable with other archival projects. In addition, we learned more about what will be necessary to make the site fully functional. In Phase II,

¹⁰ Bogan, K. and Potter, A. (2007 April 12) *archiveThing*. Retrieved April 30, 2007 from <http://web.simmons.edu/~bogan/archivething/archivething-whitepaper.pdf>

research and a business plan will be examined as well as policy building, and the technology necessary to create a fully interactive web application.

Phase II of archiveThing

Phase II will begin with additional research and the development of a business model and plan for putting together and marketing archiveThing as a functional web service. This research will also determine if the niche archiveThing serves is large enough to support the continuation of the project. Jeffery Timmons, professor of entrepreneurship at Babson College, recommends answer the following questions before moving any further on a potential venture:

1. Does the idea create or add significant value to the end users?
2. Does the idea solve a significant problem or meet a significant want?
3. Does the idea have robust marketing, margin, and moneymaking characteristics?
4. Is the idea a good fit with the founders of the venture?¹¹

Answering these questions will require more research into this area. However, archiveThing may be able to use this work to obtain grant funding for future implementation.

Next, extensive work needs to be done to build server-side and interactive elements of archiveThing so that it can move into a more functional alpha stage. A database, either relational or native XML, will be necessary to manage the various collections, users, and user-created tags. A native XML database, which is document-centric, would organize individual XML documents into a clear hierarchy, while a

¹¹ Timmons, J, Zacharakis, A, and Spinelli, S. (2004) *Business Plans that Work: A guide for small businesses*. New York: McGraw-Hill. p. 26.

relational database would allow more fine-grained control over the data within those documents.¹² Server space will have to be obtained to store and make accessible the various parts of the site. Which server language is used will partially depend on where hosting space is set up. Finally, a robust search function and strong security will have to be put in place. We will probably need to hire someone with this programming knowledge to create these back end features.

An important part of archiveThing is for it to be integrated into current online archival initiatives. In Phase II, an Open Archive Initiative, such as OAster, will harvest archiveThing.¹³ OAI's protocol for metadata harvesting (OAI-PMH) gathers metadata from multiple sources and places it in a single, searchable database. The requesting protocol is based on HTTP, and it returns results in XML, making it flexible for integration into archiveThing. OAI links to a number of software solutions to help implement their harvesting protocol.¹⁴ archiveThing also wants to supply harvested information to its users so that archiveThing can provide information from archives that are not a part of archiveThing.

Greater interactivity will be incorporated into archiveThing through web 2.0 technologies such as tag clouds, tagging, commenting, a "favorites" or bookmarking system, and possibly user blogs or a forum. At a later stage, customizable CSS for profiles will help users feel like they "own" archiveThing. Similar websites, such as LibraryThing¹⁵ and Instructables¹⁶ use these types of tool to encourage communication

¹² Bourret, R. (2005) *XML and Databases*. Retrieved April 29, 2007 from <http://www.rpbouret.com/xml/XMLAndDatabases.htm#why>

¹³ <http://www.oaister.org/>

¹⁴ <http://www.oaforum.org/tutorial/>

¹⁵ <http://www.librarything.com/> Which uses tagging and some forum activity.

and discovery between their users. The people who are likely to create a collection are also the ones who use and enjoy web 2.0 tools. Incorporating tools that allow user contribution and remixable content will enrich the user experience and make archiveThing more than just a list of items.

Finally, policies for handling copyright, privacy, access, and preservation will have to be developed to ensure a productive and safe experience on archiveThing. Copyright will have to be addressed on multiple levels. First, archiveThing will need to be copyrighted or licensed in some manner; this policy will probably be related to Creative Commons. archiveThing will also need a policy to distinguish copyright held by the institutions or individuals who own collections. archiveThing can follow the example of other sites, like the September 11th Digital Archive, that display personal images and stories. Privacy will probably be addressed on a similar level as LibraryThing, which only requests minimal information (a username and password). Since archiveThing will have unique items uploaded, an email address may also be requested so users do not lose access to their material.

Finally, preservation concerns for the uploaded collections need to be considered. The goal would be to guarantee long-term preservation of the uploaded material at archiveThing, but this is easier said than done. Even large institutional repositories, such as DSpace, can only offer to make their best effort to support some formats and at least

¹⁶ <http://www.instructables.com/> Which features a nice voting/favorites system and operates almost entirely like a forum for both its user created instructuables and its site discussions.

preserve the bit stream on others.¹⁷ archiveThing will also have to be able to address the preservation implications of people being able to remove their collections from the site; an activity not usually compatible with long-term preservation. Overall, archiveThing hopes to be a valuable tool for archivists and people interested archives without placing undue pressure on people's right to privacy or copyright.

The experience of creating archiveThing was an enlightening one. The task was a bit daunting for the time period allotted to complete the project, but it revealed a practical application of XML in archives and provided an opportunity to experiment with many aspects of XML. archiveThing has the potential to change the way archives and some of their potential patrons interact and share information and will help create a more collaborative environment for everyone involved.

¹⁷ (2002-2007) DSpace Policy Issues: What type of preservation support does DSpace provide for content? Retrieved on April 30, 2007 from <http://www.dspace.org/implement/policy-issues.html#preservation>

Appendix A: Breakdown of work done on archiveThing

This is a fairly rough break down of the work we each did on archiveThing. On all documents we both looked over the work done and made changes when necessary.

Kelli Bogan	Amanda Potter	Completed Jointly
Burns EAD type collection xml	Simmons EAD type collection xml	Archivething.xsd
Burns Description Type collection xml	Simmons Description Type collection xml	Archivething.css
Photo list type collection xml	Pirate Librarian Description Type collection xml	General site design/color selection
kbogan profile xml	Piratelibrarian profile xml	Presentation
Burns profile xml	Simmonsarchive profile xml	White paper
EAD type XSL	Profile XSL	Final paper
List type XSL	Description type XSL	
Content for about page	XHTML for browse, home, resources, and about	
Both prototype forms (XHMTL)	Graphic design (Photoshop)	
Obtained URL	Implemented tags (tag code obtained from Dynamic Drive)	
Profile XHTML mockup	Implemented browse tree (code obtained from javascripttoolbox.com)	

Bibliography

- BBC. 2006. WW2 People's War. Retrieved April 17, 2007 from
<http://www.bbc.co.uk/ww2peopleswar/>
- Bentley Staff. 2006. Polar Bear Expedition Digital Collections. Retrieved April 28, 2007
from http://polarbears.si.umich.edu/index.pl?node_id=272&lastnode_id=1163
- Bogan, K. and Potter, A. (2007 April 12) *archiveThing*. Retrieved April 30, 2007 from
<http://web.simmons.edu/~bogan/archivething/archivething-whitepaper.pdf>
- Bourret, R. (2005) *XML and Databases*. Retrieved April 29, 2007 from
<http://www.rpbouret.com/xml/XMLAndDatabases.htm#why>
- Center for History and New Media. (2005). Hurricane Digital Memory Bank. Retrieved
April 30, 2007 from <http://www.hurricanearchive.org/>
- . (2005). September 11th Digital Archive. Retrieved April 30, 2007 from
<http://www.911digitalarchive.org/>
- DLPS/DLXS. (2002-2007) OAIster. Retrieved April 30, 2007 from
<http://www.oaister.org/>
- DSpace (2002-2007) DSpace Policy Issues: What type of preservation support does
DSpace provide for content? Retrieved April 30, 2007 from
<http://www.dspace.org/implement/policy-issues.html#preservation>
- Dynamic Drive. (2006) Overlapping horizontal tabs. Retrieved April 30, 2007 from
<http://www.dynamicdrive.com/style/csslibrary/item/overlapping-horizontal-tabs/>
- Instructables. (2006) Instructables: Step-by-step collaboration. Retrieved April 30,
2007 from <http://www.instructables.com/>
- Krug, S. (2006) *Don't Make Me Think: a common sense approach to web usability*.

- Berkeley, CA: New Riders Publishing. p. 79-80.
- Kruse, M. (2006 September 18) DHTML Exandalbe and Collapsible Tree. Retrieved April 30, 2007 from <http://javascripttoolbox.com/lib/mktree/>
- Library of Congress. (2006) <ead> Encoded Archival Description Version 2002 Official Site. Retrieved April 30, 2007 from <http://www.loc.gov/ead/>
- LibraryThing and Spalding, T. (2007) LibraryThing. Retrieved April 30, 2007 from <http://www.librarything.com/>
- Notess, G. R. 2006. The terrible twos: Web 2.0, Library 2.0, and more. *Online* 30 (3): 40-42. Retrieved April 28, 2007 from <http://www.infotoday.com/online/may06/OnTheNet.shtml>
- OAI. (2007) Open Archives Initiative Home. Retrieved April 30, 2007 from <http://www.openarchives.org/>
- OAF (n.d.) OAI for Beginners - the Open Archives Forum online tutorial. Retrieved April 30, 2007 from <http://www.oaforum.org/tutorial/>
- Timmons, J, Zacharakis, A, and Spinelli, S. (2004) *Business Plans that Work: A guide for small businesses*. New York: McGraw-Hill. p. 26.