Automatic Generation of the SIGWEB Anthology CD

Sunghun Kim, Jim Whitehead
Dept. of Computer Science
Jack Baskin School of Engineering
University of California, Santa Cruz
Santa Cruz, CA 95064 USA
{hunkim, ejw}@cs.ucsc.edu

ABSTRACT
The ACM SIGWEB Anthology of Hypertext and Hypermedia is a CD-ROM incorporating ACM Hypertext conference proceedings from 1987 to 2003 and the research articles from three special issues of Communications of the ACM (CACM) on hypertext and hypermedia. The CD-ROM also includes bibliographic metadata in three formats, EndNote, Bibtex, and Dublin Core. The vast majority of original content on the CD-ROM (tables of contents, abstracts, bibliographic metadata) was automatically generated based on information extracted from the ACM Digital Library. This paper describes the process and tools used to create the CD-ROM image, as well as lessons learned. Central to the generation process is replication of ACM Digital Library articles and metadata to a WebDAV server, followed by the execution of XSLT stylesheets to transform the WebDAV metadata into HTML, EndNote, Bibtex, and Dublin Core formats. Tools employed are freely available, making it easier to replicate the described process.

Keywords
CD-ROM production, WebDAV, XSLT, BibTeX, Dublin Core, EndNote, hypertext, hypermedia, digital libraries

1. INTRODUCTION
To better promote the dissemination of research results from the hypertext and hypermedia research community to its membership, the Association for Computing Machinery (ACM) Special Interest Group on Hypertext, Hypermedia, and the Web (SIGWEB) decided to publish a compendium of the research papers from the ACM Hypertext and Hypermedia conference series from 1987-2003, including the 1992 and 1994 European Conference on Hypermedia Technology (ECHT). The 1990 ECHT conference is not included since the copyright is not held by ACM. Additionally, papers from three special issues of Communications of the ACM (CACM) were included in the compendium (July 1988, February 1994, and August 1995). The compendium, titled the "ACM SIGWEB Anthology of Hypertext and Hypermedia," was distributed to ACM SIGWEB members via a mass postal mailing of CD-ROMs.

In mid-2003, when the anthology was created, all SIGWEB members had access to the ACM Digital Library [1] (ACM DL), and hence to all of the included papers. ACM DL articles are searchable, and have links to referenced articles that are also in the ACM DL. Why then create a static snapshot of portions of the ACM DL? There are several reasons. First, by packaging the research together into a CD-ROM, it raises the visibility of the articles. Instead of being one conference series among many, the anthology specifically highlights hypertext and hypermedia research. The CD-ROM has significant access speed advantages over the ACM DL. Once the Acrobat PDF viewer software is loaded and running, accessing the *entire contents* of an article from the CD-ROM takes 3-5 seconds. While the ACM DL can equal this speed under the best of conditions (low load, fast network, small PDF file), at times it can take much longer. Additionally, many articles in the ACM DL are retrieved a page at a time, which is disruptive to the reading experience if each page takes 10s of seconds to load. There are also infrequent occasions when the ACM DL is unavailable due to maintenance, system failure, or problems with the network being used for access. Reliability of the CD-ROM isn't affected by the latest Internet worm or virus outbreak.

The CD-ROM makes it much easier to work with collections of papers. Educators wishing to use articles in their classes can more easily create selections of these by browsing and linking to articles on the CD-ROM. Researchers wishing to create a small research digital library can easily use the materials on the CD-ROM as the seed for their collection, so they can focus on adding more advanced services. Bibliographic metadata on the CD-ROM has a more uniform representation than the ACM DL, where this information must be retrieved from HTML pages, making it easier to use and perform research using this metadata.

One fear raised about the anthology was that it might cause access to hypertext and hypermedia materials in the ACM

DL to decline. This is a concern because ACM special interest groups receive revenue based on the number of accesses their material receives in the ACM DL, and for many SIGs this is a substantial portion of their total yearly income. We believe these fears are unfounded, for several reasons. The anthology is being made available to only a small fraction of the total user base for the ACM DL, and hence it's reasonable to assume that many of the current accesses to the hypertext materials on the ACM DL are by people not receiving the CD-ROM. Even for those that do receive the CD-ROM, there are many reasons why they might still elect to use the ACM DL, including rich searching of the entire digital library, and linking of references. Furthermore, the anthology CD-ROM will be out of date as soon as the next hypertext conference is held, whereas the ACM DL will still be current. Finally, people have a tendency to misplace, lose, or forget about physical CD-ROMs, while the ACM DL is always just a click away.

2. GOALS

Several goals motivated our work on the anthology. We wanted it to be as complete as possible a record of the printed materials of the ACM Hypertext conference series. We also wanted to include bibliographic metadata in commonly used machine-readable formats (BibTeX [8] , Dublin Core [14], EndNote [7]), so that people could easily and correctly reference materials in the anthology, as well as mine this metadata in future research. Materials should be accessible using a Web browser and a PDF reader, leveraging current de-facto standards. Each conference and CACM special issue should have its own table of contents page, and the abstracts for each article should also be available.

There are may other desirable features that a CD-ROM anthology might have. A master index of authors and their papers would be very nice, as would links from references to papers. Rich associative linking within the paper contents themselves would be very much in the spirit of Bush, Nelson, Engelbart and van Dam. A subject index would also be valuable. Guided tours among the contents by subject area experts would increase the value of the anthology for both new and experienced researchers. It would also be good to include the published papers from workshops associated with the hypertext conference, when copyright permissions can be obtained. Similarly, inclusion of the video proceedings from Hypertext 1991 and 1993, and video of keynote addresses would add valuable material. In the end, none of these desirable features was included, primarily due to time constraints. The anthology CD-ROM was produced using entirely volunteer labor, and this limited the amount of time that could be dedicated to the project. In the end, what was seemingly a 3-4 day project took two people 3½ weeks. The compares favorably to the approximately two months per instance reported for the 1998 Hypertext on Hypertext project, which included only nine articles [2].

An additional goal we set for ourselves was to automatically generate as much of the original anthology content as possible. This includes tables of contents for each conference or CACM special issue, bibliographic metadata files, and article abstract pages. It seemed reasonable to us that future anthology updates might be created, and we wanted to create a toolset that would dramatically reduce the time required to make them. As well, it seemed reasonable that other SIGs might want to create similar anthologies, and would also benefit from these tools.

3. METHODS

Two observations underlie the process used to generate the anthology contents. First, the articles and metadata already exist in machine-readable form in the ACM DL, albeit embedded in HTML pages. Next, we noticed that the automatically generated content is primarily format permutations of the bibliographic metadata. A table of contents provides one view of this data, represented in HTML, while a Bibtex file includes the same data, in Bibtex format. Based on this, it seemed reasonable to assume that if the metadata could be stored in a known format in a content management repository, it should be possible to write converters to translate this metadata into multiple formats.

In a nutshell, the generation process reads articles and bibliographic metadata from the ACM DL, and then stores the articles as resources on a WebDAV server [15,6]. Bibliographic metadata is stored as WebDAV properties on the resources. A shell script then invokes the Xalan XSLT processor [3] multiple times to convert the XML output of the WebDAV property retrieval method (PROPFIND) into HTML, Bibtex, EndNote, and Dublin Core formats. Figure 1 shows an overview of this process.

The first and most tedious step in the process is crawling and gathering content data and metadata from the ACM DL, encoded in HTML. One problem encountered was non-

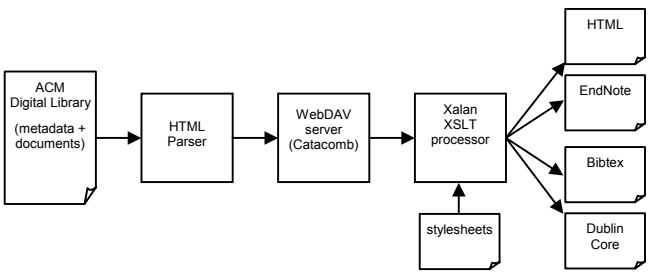

Figure 1. Process for automatically generating contents of the anthology CD-ROM.

uniform HTML pages. The table of contents for the ACM Hypertext conferences are the same, with the exception of HT 2000 which was different enough to require a specialized HTML parser. Additionally, the table of contents for the three CACM special issues differed from the HT conferences. Within HTML files using the same structure, there were occasional errors in the content. For example, the July 1988 CACM special issue had commas instead of spaces in some author's names, e.g., "Frank,G. Halasz." This required special-case parsing.

We created HTML parsers for the different TOC styles using "El-Kabong HTML," a speedy HTML processing library [13]. To retrieve the PDF article content data, we used GNU wget [5]. Even here we had to be careful, since under high load conditions the ACL DL returns an HTML error page stating that the server is busy, and to try again. We had to test for this condition to ensure that we successfully retrieved all article contents.

We chose a WebDAV server as the content management repository used as the intermediate store for articles and their associated bibliographic metadata. While a DBMS could have been used, storing content data and metadata simultaneously is hard using DBMS. PDF article content data can be stored using BLOBs or a field that references a file in the filesystem. The former way increases the size of the database, and runs the risk of exceeding BLOB size limitations (less of an issue with recent versions of most DBMS's, such as MySQL) [10]. The latter way uses two separate systems, DBMS and a filesystem to store content data and metadata, requiring procedures to ensure the two are kept consistent. Finally, having gone to the effort of making a DBMS have the desired features, it is at that point essentially a simple content management system.

WebDAV provides functionality to store content data as a resource, set arbitrarily structured XML metadata on these resources, and then execute searches over the article metadata. The result of these search requests is formatted in XML, thereby permitting the use of the rich set of existing tools that parse and manipulate XML.

We chose the Catacomb WebDAV server [12] because it is based on the popular Apache mod_dav module, and additionally uses a relational database as its underlying physical store. Catacomb also supports the DAV Seaching and Locating (DASL) protocol, which we used to submit queries over the bibliographic metadata. Finally, since Catacomb was developed in our lab at UC Santa Cruz, we already had considerable expertise and confidence in this technology.

One issue we quickly encountered was the lack of a tool that could execute WebDAV commands from within a shell script. The well-known Cadaver client [11] provides a command-line interface to a WebDAV server, but is not well suited for script execution. To address this need, we created Davtool [9], a WebDAV method execution tool that operates similar to wget. Davtool takes as input a WebDAV method, URL, request body file, and authentication credentials, and then executes the specified method, saving the response message body in an XML file. We used Davtool to perform the SEARCH method, thereby retrieving article metadata into an XML file. The XML metadata file contains all of the metadata for a single conference or CACM special issue.

With the metadata in XML format, we were able to use the Xalan XSLT [3] processor to automatically convert it into a range of text and XML formats. Individual XSLT style sheets were written to covert metadata into Bibtex, EndNote Import (Refer), and Dublin Core metadata formats. Another stylesheet created the TOC page for each conference and CACM special issue. Each article abstract is displayed on a separate page, and each of these pages is created by one stylesheet, executed once per abstract. Since XSLT does not permit multiple output files to be created by a single stylesheet execution, we used a stylesheet to create a shell script that, in turn, called Xalan once per article to create the HTML file containing its abstract.

Once the articles and metadata were correctly added to the WebDAV server, complete execution of the CD-ROM generation process takes approximately 10 minutes on a 2 GHz Pentium Linux machine. Compiled stylesheets (using XSLTC) could potentially increase this speed, an option we did not explore. XSLT processing was by far the most time consuming aspect of the generation process.

4. LESSONS WE LEARNED

In this section, we discuss some of the issues and lessons learned from creating the anthology CD-ROM.

4.1 Crawling

Crawling and data gathering from HTML web pages is one of the most tedious aspects of the entire process, due to the combination of non-uniform HTML structures and occasional data quality problems. Due to these problems, we couldn't write a generalized HTML parser to download articles and metadata from the ACM DL. Making article metadata available in a machine-readable format such as an XML representation or an enhanced form of RSS [4] would make it possible to use a general-purpose parser to retrieve this metadata. However, it would also make it much easier for someone to replicate and use this material without obtaining permission.

4.2 Internationalization

Many papers have authors whose names include characters in the ISO-latin character set (e.g. Kaj Grønbæk). Unfortunately, representations of these characters in HTML, XML, Bibtex, and EndNote are all different. For

example, “Grønbæk” is “Grønbæk” in HTML, “Gr{\o}nb{\ae}k” in Bibtex, and ISO-latin encoded in XML and EndNote. Luckily, Xalan automatically converts ISO-latin into the correct encodings for HTML, and preserves ISO-latin encoded characters for XML and text (EndNote). However, for Bibtex we wrote a utility to convert ISO-latin into Bibtex characters, since Bibtex does not correctly format ISO-latin characters in printed output.

4.3 Author Names

Handling of author names was surprisingly tricky, in large part because names themselves are very non-uniform. First, parsing names to determine the first name, middle name, last name and suffix was difficult, and involved writing several special case handlers. Even so, there are still some exceptions that need to be handled manually. Additionally, at first it wasn’t clear how the name information should be represented. We initially stored the first, middle, last, and suffix information separately. However, when using XSLT, it was much easier to embed the entire name as it originally appears, rather than writing XSLT statements to manually reassemble names from constituent pieces. It is tedious to handle all of the permutations of absent first, middle, last and suffix information and still get the intra-name whitespace correct.

4.4 Article Naming Convention

One surprise was the failure of our original article naming convention. Initially, we used the convention of {last name of first author}{year}{first meaningful word of article title}, one example being “Halasz1994Dexter.” The problem with this convention is that names are not always unique within a given conference. One instance occurs in the Hypertext 1993 conference, where Wendy Hall is the first author of an article describing a clip in the video proceedings, and she is also the first author of an article accompanying a technical briefing. Both articles start with the word “Microcosm,” the name of the system being described, and hence both articles would have the name “Hall1993Microcosm.” While it was conceivable to add an instance identifier to article names, this would have required additional special-case code, and so we decided to use the p{first page number}-{last name of first author} convention employed by the ACM DL. This solved our problem, since the page numbers uniquely identified the two pieces by Wendy Hall.

4.5 Data Quality

Unfortunately, the ACM DL contains missing and incorrect information. One issue is that capitalization of paper title differs between the HTML and PDF files in the DL. In this case we generally capitalized each word except prepositions and articles. This involved writing a special-purpose converter to repair article titles, but raises the possibility that we introduced additional capitalization errors.

Article abstracts and keywords are not uniformly available for all HT conference proceedings. It is possible to repair this, since the digitization process used to scan articles to PDF also performed optical character recognition on the text. As a result, abstract text can be cut-and-pasted from scanned articles. However, there are many errors in such text, and it requires much human supervision to get this right. We used this technique to recover the article abstracts for the Hypertext 1991 conference, manually extracting and correcting the abstract text. The entire conference required about 6 hours of work to recover the abstracts. Due to time constraints, we did not recover abstracts from any other conference.

A final note on data quality is that we still do not have a good mechanism for ensuring that there are no errors in the generated content, or that all of the retrieved articles are uncorrupted. We have so far been relying on manual inspection, with the likely outcome being errors in the final anthology.

4.6 XSLT

While we were generally impressed by the ease with which XSLT permitted the construction of custom converters, we did encounter a few frustrations with the expressiveness of the XSLT language. We are happy to note that all of these problems have been addressed in the new XSLT 2.0 standard.

First, handling page ranges was difficult in XSLT. Page ranges are expressed in the form {number}-{number}, where the dash and second number may not be present in the case of single page articles, like panel overviews. Unfortunately, the built-in string handling capabilities of XSLT return a null string if a substring search condition is not found. So, the operation to return the string before the “-” would return the first page in a range (e.g, “5” for “5-15”), and the null string for single pages (e.g., null for “16”). The workaround was to convert the dash to a space, and then use the string to integer conversion function, which ignored everything after the space.

Another issue with XSLT is the difficulty in correctly handling whitespace. The source of generated HTML files do not look very nice due to this. While it seems that XSLT could be made to format the HTML correctly, it would require additional fiddling and time to do so.

The design of the XSLT language inherently combines logic and presentation. While this makes it easier to initially write XSLT code, these programs are most likely difficult for others to understand, and it is possible there will be maintenance issues for this software as a result.

4.7 Structure

A non-issue for the anthology was the structure of the contents on the CD-ROM. We used a conventional approach, with a top-level directory containing the anthology's table of contents, and with a series of sub-directories, one for each conference and CACM special issue (17 total). Within each directory there is a table of contents page that lists all of the articles within, mostly mirroring the table of contents of each conference proceedings or CACM special issue. We didn't consider alternate structures, in part due to the time that would be required in such a restructuring, and in part because we felt satisfied with this structure, and were not compelled to change it.

A few observations can be made about the structure. First, it is mostly designed to replicate the structure of the printed media from which the materials are drawn. It is conceivable that materials could have been organized along subject lines, or by author, or strictly chronologically (mixing CACM contents with conference contents in those years where both occurred). Nelson's critique of hierarchy certainly applies here. Any of these hierarchies would have worked well, and it's difficult to construct grounds for choosing one hierarchy over another. But, by picking one hierarchy and burning it into the CD-ROM, we are privileging one view of the information, to the detriment of others. Nevertheless, the current structure does seem to mirror one way that hypertext researchers internally organize the literature, since it mirrors the printed form.

We did not encounter the same set of structural issues as did the 1988 Hypertext on Hypertext project [2], in large part due to the use of PDF. The previous project had to wrestle with the best way to covert printed articles into the chunk-oriented hypertext systems used in the project (HyperCard, HyperTies, and KMS). All issues concerning humans reading electronic documents in the current anthology are delegated to the PDF reader. As a result, we did not need to worry about creating tables of contents within articles, develop page flipping conventions, or breaking articles into text chunks. PDF documents are page-structured scrolls, and hence we are in the "holy scroller" camp in the old debate between cards of text and scrolls of text.

5. FUTURE WORK

Many future activities could be based on the current CD-ROM. It would be interesting to construct of a network of associative links among the documents, highlighting commonality and development of ideas over time. One issue raised in the construction of the 1998 Hypertext on Hypertext project was the difficulty of creating non-trivial associative links among the articles in that collection [2]. We suspect that we would not encounter this difficulty when linking among the current anthology, since the larger set of documents implies there are many connections to be made. In a similar vein, having guided tours among the materials would also be quite valuable.

Several issues need to be addressed to make this possible. Creating large sets of associative links or substantive guided tours is a time consuming piece of scholarship, but there are currently no mechanisms to reward such work. Perhaps an online journal could be established to accept and peer review networks of associative links and guided tours.

An additional issue is the form and format of links and guided tours. The current use of PDF formatting for documents implies that only the current linking and annotation capabilities of the Acrobat reader can be used. However, these do not well support the needs of an academic community. While annotations can be stored externally to PDF documents and overlaid when the document is viewed (using WebDAV-based collaboration features), there is no support for selecting among multiple sets of overlays within the Acrobat reader. This capability would be necessary, since over time multiple link sets would be created, and a reader would not want all link sets to be active at the same time, to avoid being overwhelmed by links and distracted by link markers. Additionally, there are no facilities for having a guided tour lead a reader through multiple documents.

Given this state of affairs, two possible paths present themselves. One would be to transition away from PDF into a document format that had a more clear separation of content and presentation, such as plain text, or XML plus XSL-FO stylesheets. Figures would need to either be converted to bitmaps, or perhaps translated into vector graphics using a standard such as SVG. While this would open up the literature for a wide range of additional capabilities, the cost of translation is, with current technologies, quite large. Alternately, it might be possible to develop a set of concrete recommendations for improvement to the Acrobat reader itself, and then use these recommendations to engage engineers at Adobe. Given the prevalence of PDF use for digital libraries today, it's reasonable to assume that Adobe might want to better support this use of the Acrobat reader.

6. CONCLUSION

Several conclusions can be drawn from our experience creating the anthology CD-ROM. Reading the articles and metadata from the ACM DL was surprisingly difficult, since the impression of uniformity conveyed by a visual observation of its Web pages is different from the non-uniform nature of the actual HTML structure. Using a WebDAV server to store articles and bibliographic metadata worked very well, and its native use of XML to represent metadata search output made it easy to use XSLT

for content generation. While XSLT processing was slow, and it was at times tedious to create correct stylesheets, we have no doubt that it was easier to write an XSLT stylesheet than a corresponding program in Java or C to accomplish the same conversion function.

We hope the method we used for this CD-ROM can serve as a reference method for hypertext researchers, and other academic communities. Ideally others who wish to create a similar compendium CD-ROM will reuse this method.

All tools mentioned in this article are either available on the Web, or by request from the authors.

ACKNOWLEDGEMENTS

The Cascading Style Sheets used on the CD-ROM are based on the BlueRobot.com "Left Menu" style.

REFERENCES

[1] ACM, "ACM Digital Library," (2003). http://www.acm.org/dl/.

[2] L. Alschuler, "Hand-Crafted Hypertext--Lessons from the ACM Experiment," in *The Society of Text*, E. Barret, Ed. Cambridge, MA: MIT Press, 1989, pp. 343-361.

[3] Apache Foundation, "Xalan-Java Project Homepage," (2003). http://xml.apache.org/xalan-j/index.html.

[4] G. Beged-Dov, D. Brickley, R. Dornfest, I. Davis, L. Dodds, J. Eisenzopf, D. Galbraith, R. V. Guha, K. MacLeod, E. Miller., A. Swartz, and E. v. d. Vlist, "RDF Site Summary (RSS) 1.0," (2000). http://web.resource.org/rss/1.0/spec.

[5] Free Software Foundation (FSF), "GNU wget Project Homepage," (2002). http://www.gnu.org/software/wget/wget.html.

[6] Y. Goland, E. J. Whitehead, Jr., A. Faizi, S. Carter, and D. Jensen, "HTTP Extensions for Distributed Authoring -- WEBDAV," Microsoft, U.C. Irvine, Netscape, Novell. Internet Proposed Standard Request for Comments (RFC) 2518, Feburary, 1999.

[7] ISI ResearchSoft, "EndNote Homepage," (2003). http://www.endnote.com/.

[8] D. Jacobsen, "The BibTeX Format," (1996). http://www.ecst.csuchico.edu/~jacobsd/bib/formats/bibtex.html.

[9] Sung Kim, "Davtool Project Homepage," (2003). http://davtool.sourceforge.net.

[10] MySQL AB, "MySQL: The World's Most Popular Open Source Database," (2003). http://www.mysql.com.

[11] Joe Orton, "Neon HTTP and WebDAV Client Library," (2003). http://www.webdav.org/neon/.

[12] Kai Pan, Sung Kim, Elias Sinderson, "mod_dav_dbms: A Database Backed DASL Module for Apache," (2002). http://www.webdav.org/catacomb/catacomb_arch.pdf.

[13] J. Travis, "El-Kabong HTML Project Homepage," (2002). http://ekhtml.sourceforge.net/.

[14] S. Weibel, J. Kunze, C. Lagoze, and M. Wolf, "Dublin Core Metadata for Resource Discovery,". Internet Proposed Standard Request for Comments (RFC) 2413, September, 1998.

[15] E. J. Whitehead, Jr. and Y. Y. Goland, "WebDAV: A Network Protocol for Remote Collaborative Authoring on the Web," *Proc. Sixth European Conference on Computer Supported Cooperative Work*, Copenhagen, Denmark, Sept. 12-16, 1999, pp. 291-310.