Personal Libraries:
Collection Management as a Tool for Lightweight Personal and Group Document Management

Robert Wilensky
Division of Computer Science
University of California, Berkeley
Berkeley, CA 94720
December 29, 2000

1. Introduction

The traditional model of scholarly information dissemination works like this: Originators create works, which they submit to publishers. Publishers filter these submissions, generally assembling those works that pass through the filter into aggregates (i.e., journals), which they then make available. Libraries select and assemble these aggregates into collections, which they maintain and organize. Libraries make the collections accessible to users, and also, attempt to preserve what they have assembled.

The division of labor into originators, publishers, libraries and users doesn't transfer cleanly to the digital world. For example, the CACM Digital Library ([1]) is a collection of journals, etc., published by CACM. I.e., it is run by the publisher, and hence has no librarian, in the sense of separate "downstream" selector and aggregator. The Los Alamos preprint server ([5]) provides a hosting service for originators, leaving out publishers and librarians (in the sense that no selection is performed). NCSTRL ([6]), the distributed computer science technical report collection, is a self-federating set of services, each independently maintained by various research institutions. These institutions represent the originators of the information (i.e., the computer research laboratories), leaving no role for a publisher or a librarian. Some organizations play a variety of roles. For example, the California Digital Library ([2]) hosts some collections, and federates others, serving in this capacity as a indexing service to documents hosted elsewhere, and as a licensing service to its users. And, of course, individuals can simply place a document on an available web-server, with access facilitated by search services, again leaving out publisher and librarian.

While, as these examples indicate, there is a diversity of models for digital libraries, in general, the technology offers the possibility of dis-intermediating traditional "content middlemen", i.e., publishers and librarians, and hence, resources are held closer to the originator. Thus the CACM digital library holds resources at the publisher, Los Alamos at a server on behalf of the originator, and NCSTRL (and the unadorned web) at the originator. We view this direction as natural and positive, since it enables greater flexibility and perhaps much more efficient economic models ([9], [12]).

Here we are concerned with several issues that arise in this transition. As the previous discussion suggests, there are many models for digital libraries in operation. It would be

useful to have an overall architecture for digital library services in which these can be described and analyzed, and improvements found. We present such an architecture. This architecture differs from other proposals, e.g., [4], primarily in the separation of what we will call collections, i.e., metadata about sets of documents, from repositories, i.e., the mechanism for hosting the documents themselves. Given the separation of collections from repositories, the issue of collection management services arises. This is an ostensible new form of service, and is the primary focus of this report. Among other things, collection management services should make it easy for individuals and groups to create and maintain collections. Thus, such services facilitate the construction and management of lightweight personal and group libraries. Personal libraries allow users to create collections of distributed documents with the functionality and discipline of full-fledged library services, yet at the same time, being lightweight, at cost comparable to managing resources in a file system or at a web site.

Personal libraries are where digital libraries and personal information management intersect. In effect, we want to make available to each individual the services that one can expect from a full-fledged digital library, however constituted. One crucial aspect of personal information management, we have found, is incorporating legacy (i.e., paper) documents into the digital system. While this is properly an issue of repository rather than collection management, both are required for a useful system, so we discuss acquisition of scanned documents as well. We discuss a prototype implementation of a collection management system, and our associated repository service. Finally, we discuss variations on this theme, namely, repositoriless collections, and the interaction of personal libraries with another development in which we are engaged, self-administering documents ([3]).

2. Overview

Previously, our project developed an on-line collection of documents with a specific purpose in mind ([14]). Most of these documents were created from scanned page images, via a custom process. In this case, the documents existed previously only on paper, and hence, we hosted them in our own repository. We provided full-text and meta-data indexing for this collection, plus other, more experimental services ([11], [7]). A few documents existed in electronic form; some of these were hosted by us, but others already had homes elsewhere.

Having available in digital form documents that previously only existed in paper, unsurprisingly, turned out to be quite useful to our user base, especially when these documents were integrated with digitally-created documents. Indeed, the utility of such assemblies readily became apparent for many other applications, which were essentially of personal or group information management nature. For example, consider the process of creating a course reader. In the paper-based world, one identifies the documents one would like to include, and then creates a set of copies of them, which is given to a copy service. The service requests from copyright holders' permission to duplicate the articles, and a reader is photocopied. In the digital world, we could scan these documents to produce on-line versions in some useful format; then these on-line versions can be

integrated with digitally available versions of some papers. The same digital library services available for large collections would enable students to access, browse, search and annotation this collection, and so on. Many other motivations for pulling together diverse collections have since arisen, from assembling literature surveys of a field to customizing a single body of material for different cohorts.

We began making customized such collections by hand, though through a rather cumbersome process, as a system administrator had to set up each collection on an ad hoc basis. Setting up a collection was not trivial for the administrator, as each collection may have its own access conditions. Moreover, integrating scanned image documents into a collection involved more specialized system administration. Thus, while the idea of specialized collections was valuable, it was quite clumsy to execute. Hence, we decided to build an infrastructure that would support such rapid customization..

The infrastructure conceptualizes services as belong to one of three kinds: Collection management services, repository services, and document-level services. Repositories simply house documents; i.e., they perform the essential services of incorporating and disseminating documents, although they may perform other services as well, such as version control. Many document management services available today are essentially repository services of one form of another. In contrast, collections are virtual aggregations of documents. The documents themselves do not reside in a collection, but live elsewhere, e.g., in some repository. The collection is in effect just the document metadata, along with the proper protocol to access the actual document. A collection management service provides authorized users with the ability to create, populate, edit, and otherwise manage collections; it also constructs indices to let authorized users search, browse and otherwise access collections.

While collection managers are separate from repositories, they may work together. For example, a collection management service may also provide repository service through some repositories with which it has an association. Such repositories may provide local copies of documents for the purpose of availability, or provide hosting of documents that have no other home.

Finally, document-level services are services that apply to documents as individuals. Examples of such services include language translation, format conversion, and delivery. Repository services and collection management services may support the application of such document-level services to documents they manage, but of course, the services themselves are outside the scope of repository or collection management per se.

3. Example

Here is a simple example to illustrate how the model appears to users. A user approaches a collection manager service (which may require the user to authenticate him- or herself). The user then selects desired services, access to which will be provided given proper authorization. For example, the user might want to browse or search a collection or collections hosted by the manager. More interestingly, the user might attempt to create a

new collection. If authorized to do so, the user will be able to supply a collection name (e.g., “cs294-reader”), and some access conditions for various collection services—for example, the user may want to allow TAs to edit the collection, and students only to access it—along with other collection metadata. Then the user may begin populating the collection. Doing so means supplying the collection with metadata for individual documents. The user may want to make the document available in multiple formats, so several ways to access the document may be provided by the user. In addition, the user may want the document to exist in a repository. In this case, the collection manager asks an affiliated repository to incorporate the document. The incorporation process itself may do considerable work. For example, if the document is a set of scanned page images, incorporation will involve converting the images to a format appropriate for viewing, performing clean-up operations on the images, and running an OCR process on them.

Alternatively, the user may want to add a new format to an existing document, or remove a document from the collection. The user may also want to provide style sheets for presenting the results of browsing the collection to various classes of users. For example, the user may want hide some metadata from students in the course, but not to TAs, or others who might also have collection editing privileges.

Most users approaching the collection manager will simply have privileges to access given collections. Documents added to a collection will immediately be come browsable and searchable by metadata. In addition, at some implementation-dependent point, the collection manager creates or updates a full-text index of this collection. Authorized users can then perform full-text searches on the given collection, or on all the collections hosted by the collection manager for which they have access privileges.

Note that accessing a collection is quite different from accessing documents in a collection. In the first case, a user with privileges to access a collection simply sees the metadata about the document that the collection provides. In trying to access a document, the user is directed to a repository containing the actual item. Such a repository may have its own terms and conditions of access. At the moment, no additional restrictions on document access by collections is envisioned.

4. Services in Some More Detail

Here we describe the types of services discussed above in some more detail. Most of these we expect the reader will find familiar, and so we list them below under the class of service that support them. However, there are some features that we feel are crucial and which are not otherwise widely available. We discuss these first.

4.1 Useful Collection Management Features

There are two features that we believe collection managers should support. Since collections are separate from repositories, however the two are construed, it is possible that they will change in an uncoordinated fashion. In general, we take the position that “permissive, but robust” digital library systems are an attractive alternative to the more traditional “strict, but fragile” systems, and have proposed robust hyperlinks ([8]) as a

technology to provide just this robustness. Robust hyperlinks add to a URL a signature, i.e., a small piece of document content that serves to more or less uniquely specify the document from among all the documents on the web. Signatures are passed to search engines upon link failure to find new or alternative locations for a document. Therefore, when a document is added to a collection, the reference to its resource should be made robust. In addition, when that link to the document is followed, but the document is no longer at that location, the manager should perform robust hyperlink de-referencing. That is, the result of clicking on a collection link should return the result to the collection manager, which would normally re-direct the result to the user, but in the case of a broken link, would perform signature-based referencing using the robust hyperlink, and hence, offer the user possible new locations for the missing document.

The second useful feature is support for composite documents ([13]). Many type document types may comprise multiple resources. For example, scanned page image documents contain multiple resources for each page (e.g., a GIF image and the OCR process output); and some HTML documents come as multiple, conveniently-sized HTML pages. The collection manager needs to treat these multiple resources as a single coherent document, for example, creating a single index for the whole document from all its component parts. A repository may also need to know which components comprise whole documents for some tasks, if for example the document as a whole is to be deposited there, or for services liking printing or document delivery. Composites provide the capability of specifying a set of resources that together comprise a document, so they provide a relatively principled and straightforward way of accomplishing these goals.

Note that a collection may refer to multiple formats of a single document. While it is not necessary to use them in this capacity, it is straightforward to define composites that define all formats of a given document. Implementations may therefore find it convenient to structure their collections out of composites, each of which refers to all available versions of existing documents, some of which may be composites referring to multiple document resources.

We now discuss the overall set of service comprise collection management service, repository service, and document-level services.

4.2 Collection Management Service

A collection management service provides authorized parties with the means to create and manage collections. More specifically, the following services are provided:

- Collection Management Services:
  - Collection creation: Create a collection at the request of an authorized user
  - Collection editing: Adding and removing documents, document formats, and editing document metadata. Collection managers may provide efficient ways to “populate” collections, e.g., importing another collection in its entirely, or importing anconverting metadata in a different format. In addition, upon adding a document resource, collection managers may compute signatures for robust hyperlinking.

Collection access authorization: Authorize a user to perform some collection management function.

Collection structuring: Provide some additional level of structure upon the members of a collection, e.g., providing "thread" support for document annotations.

Collection result set presentation specification: Associate style sheets with result sets for classes of users.

Searching and browsing for collections: Help user find collections hosted by the manager meeting certain criteria.

Document repository delivery: Request storage for a collection item in an associated repository.

At any one time, a collection management service is hosting a number of collections. Thus, some collection management services can be best thought of as collection services. These are the services probably most centrally viewed as digital library services, and include the following:

- Collection Services:

  Document Searching: Search a collection or collections for documents matching criteria

  Document Browsing: Browse a collection or collections, e.g., by author, title, etc.

  Result Set Presentation: Apply a registered style sheet to the result of a search or browse, in accordance with some criteria (e.g., user type).

  Robust linking de-referencing: Upon failure of a repository to delivery a document, use robust hyperlink signatures to attempt to find a version at an alternate location.

  Support for document-level services: Support document-level services directly (see below).

4.3 Repository Services

A repository is a service that hosts a set of documents. Repository services include the following basic services:

Incorporation: Uploading a document into a repository, along with possible conversion to make it suitable for the repository. For example, a multi-page scanned image document might be converted to a format for viewing, have OCR run on it, etc.

Dissemination: Emission of a document from a repository to an authorized requestee.

Repositories may also support additional services, prominent examples of which are the following:

Data replication: Make copies of the data at affiliated repositories (e.g., for caching purposes)

Data archiving: Produce an archival copy of the document to promote persistence.

Repositories may provide other services as well, such as support for version control, and local search and browsing capabilities. In general, such services will only be of indirect value to users. This is because we assume that users will want to look at collections,

whereas the physical assembly of documents on a given repository will generally be uninteresting. Of course, a repository may happen to be organized around a given theme, but our view is that any such utility should be made available through the collection abstraction, hence insulating users from repository details. Similarly, a user may search a collection to find a document at a given repository; then the user can communicate with the repository privately for special services, such as retrieving a previous version of a paper.

Note, though, that collection managers may want to take advantage of such services. For example, suppose that a repository has automatically converted a document to a new format, or otherwise modified itself, so that references from collections to the old format are now dangling. If the collection manager supports robust hyperlinking, and it is aware of the repository' s search service, it can use the robust signature to query that repository's search service, hence finding the new version of the document, if it has been indexing, even if no entry for that document exists in any global search service.

In addition, repositories may make service guarantees. The following are typical examples:

Availability: Authorized document requests will be honored within a certain time interval.

Persistence: Hosted documents will continue to be available for some time period

Preservation: Archival forms of documents will be preserved indefinitely.

Replication: A copy of an updated document from "repository of record" will be made to cache repositories.

Of course, implementing a given service guarantee can be quite complex. For example, to implement preservation, the repository may need to know when to apply a document-level conversion service (i.e. when support for a previous format comes into doubt), have an appropriate document-level service available, and then apply its own data archiving service to the result.

4.4 Document-level services

Document-level services apply to individual documents, rather than collections per se. Collections may support such services for their documents. Examples of such services are as follows:

T&C enforcement: Enforce the terms and conditions associated with the document

Self-administration property enforcement: The repository may implement a self-administrating document handler, as discussed below.

Delivery: Deliver a document to a user in some fashion. E.g., in addition to standard document delivery protocols, users may find it convenient for a document to be delivered by fax, email, or some other mode of transport.

Conversion: Convert a document from one format to another.

Printing: Apply some printing service to a document

Summarization: Apply some summarization algorithm or service to a document

Translation: Apply some translation algorithm or service to a document

Services in all cases may be provided differentially. The following classes of service users are useful to distinguish:

> Authorized collectors: A collector authorized by a collection management service has the ability to create a collection, to populate and edit the collection, and the authorize users to access the collection.
>
> Authorized users: A user can be authorized by a collector to available himself of particular services of that collection.

Thus, collectors may be able to edit collections; authorized users can access certain collections, create new collections, or upload a document to a repository.

5. Implementation

We have implemented an initial version of a collection manager and affiliated repository and document services. These tools, which we present to users as a "Personal Library" service, are available for experimentation at http://elib.cs.berkeley.edu/pl/. The Personal Library' s collection manager allows users to search individual collections or all the collections it hosts, by metadata. We are planning to add full-text search capability. Internally, the collection manager will maintain one full-text index for the entire set of managed collections, and searches all of these upon queries, hiding from users results from collections to which they do not have access.
Authorized users may start new collections, for which they currently supply a password for subsequent collection management authorization. Authorized users edit collections by adding new documents, or new formats of an existing document. In adding a new document, users will be requested to supply metadata (title, author, date, collector, etc.) The metadata is indexed for subsequent retrieval.

In addition, when adding a new document or document format, the user is asked if the reference should be to a repository service that already exists, in which case the user supplies a URL, or if the user is requesting repository services as well. If the later is the case, the user specifies the type of document to be added. The possibilities now include single HTML document, multiple part HTML document, PDF, and scanned page images. Document components may be identified via URLs, if they exist, or via a file browser. In the case of scanned page images, users may identify either a single, multiple page TIFF file produced by a scanner, or a directory of individual TIFFs produced by a scanner. In all cases, the components are incorporated into the repository. In the case of scanned images, doing so involves a rather detailed process described below. In all cases, however, ASCII text is extracted from the document to be used for full-text indexing.

Currently, no style sheets are used for collection contents presentation, and result sets are presented in a standard format.

6. Scanned Document Ingestion

As suggested above, we found it crucial to be able to easily incorporate digital- and paper-born documents into a collection. There are two parts of this process, one reasonably mature, and currently implemented by the Personal Library; the other is in the process of being implemented.

The first part involves automatic processing of scanned document images. As mentioned above, our repository service has an incorporation process that ingests scanner output in TIFF image. Specifically, this incorporation process has the following steps: The TIFF images are passed through a set of programs, from Xerox PARC[1], which deskew the images and perform certain "image dry-cleaning" to clean up noise on the pages. Then the images are run through Scanworx OCR software ([1]), which produces a file containing positioned ASCII interpretation of the images. Then the images themselves are converted to 75dpi gray scale, a size and format appropriate for on-screen viewing.

The second part of the scanned image ingestion process involves making scanning itself easier. In this regard, we have acquired networked scanners from Hewlett-Packard, which make it easy for users to scan sizeable documents and have the results placed in a location from which they are automatically incorporated into a repository.

7. Discussion

Separating collections from repositories is a very simple idea. Yet we believe it provides an interesting data management tool. While we have not yet had enough experience to make confident pronouncements, we have already found personal libraries reasonably easy to use. For example, we have already experimented with personal libraries for course readers, and now find this service indispensable. Moreover, the overhead of using the collection manager is small enough that it seems worth using for routine document management tasks, in preference to file systems or vanilla web pages. For example, the author has created a collection for his own papers, moving some to the associated repository, leaving others in other repositories (e.g., a NSCTRL server, the CACM Digital Library). The process of creating the collection is a bit time consuming, since metadata must be specified for each document. However, the resulting system is a considerable improvement on the previous hodgepodge, in which were variously stored in file systems, HTML serves, and other random locations.

How does having a collection manager compare to simply having a web page for a given topic? I.e., instead of specifying document metadata to a collection manager, one can simply create a web page with links to the equivalent "repositories". There are several answers. First, although our experience in using the PL is still quite limited, and the interface preliminary, it is probably no more work, and perhaps somewhat less, for a single user to maintain a collection than a web page, if only because one may use a user interface that takes advantage of the structure of collections and their entries, rather than edit an arbitrary web page. Second, because collections are hosted by a service, it is much easier for multiple, distributed users to maintain a collection than a given web page,

[1] Written and made available to us by Dan Bloomberg

at least in a standard file system. Third, once the collection exists, much more functionality is available: Users can search and browse through the collection, and through multiple collections simultaneously. Fourth, some types of document formats, for example, our scanned page image format, are not something the individual users could otherwise create for themselves, but with the Personal Library service, these are easy to create and share. Fifthly, once a document is in a collection, it is much more straightforward to automate other tasks, such as conversion, etc.

Each document in a collection is represented by a set of data, presented to the world as a single web page, that shows document metadata and points to available formats of the document. Having such a page is itself useful. For example, suppose a document were available in some form, say, PostScript, which is not commonly indexed by web search engines. The collection manager could extract ASCII from the PostScript, and then include the ASCII in a META tag, and then report the document "home" page to a search engine for indexing. Without the document home page, it would be difficult to make the document available. Of course, one only needs a second referring page to implement this idea, but the Personal Library conveniently provides one.

Probably most useful of all, though, is that having a logical place where a collection can be assembled seems to be quite valuable. Perhaps disciplined users could construct logical collections without these tools, but the tools seem to facilitate doing so.

8. Future Directions

There are several directions we are investigating in this work. Perhaps most interesting is the idea of "repositoriless collections". Rather than building collections out of more fundamental repository services, the idea is that collections refer to documents by names ([4]). Names are de-referenced via a name-resource resolution service to find a "nearby" copy of the document. The resource may refer to a set of Napster-like cooperating caches/services that propagate copies around to their neighbors. An infrastructure to support such repositoriless collections is currently under development.

We have also been experimenting with developing repository support for images and other data types, including geographic data. Support for adding images to an image repository, and metadata for image collections, has been implemented for a restricted class of individuals who maintain the large image collections at our site.

We plan to add support for composite documents, and robust hyperlinking, as described above.

The user interface to the collection manager has undergone several revisions, and is still in need of work. In general, this part of the design requires the engagement of a substantial user population, which is one of our major short-term goals. In particular, we plan to make collection management services and networked scanning to repositories available to members of our department for routine use in the coming months. Finally, personal libraries interact with another project in the works, self-administrating

documents ([3]). Self-administrating documents are documents that have a piece of meta-data attached to them that declaratively specifies how that document should behave. This meta-data is interpreted by a local "SA-handler". The description typically specifies how the document should move about, who has access to it, and some other aspects of collaborative workflow, all of which is carried out by a network of SA-handlers, to, in effect, create a peer-to-peer document management system.

Self-administrating documents provide a link between personal libraries and document creation. Thus, an individual can use SA handling to collaborate with others to create a document. When it is finished, the SA metadata can be modified to specify inclusion of the document in a repository and, perhaps collections. Assuming that the repository, etc., has SA handler access, updates to the document would automatically get propagated to the repository. Similarly, an SA handler accessible to the network scanner would manage the incorporation of scanned images into the target repository and/or collection manager. We plan to provide such SA handler support for both scanned image acquisition and for repository service.

Acknowledgement

This research has been sponsored by the National Archives and Records Administration and Advanced Research Projects Agency/ITO, "Intelligent Metacomputing Testbed", ARPA Order No. D570, issued by ESC/ENS under Contract #F19628-96-C-0020.

References

[1] The CACM Digital Library, http://www.acm.org/dl/.
[2] The California Digital Library, http://www.cdlib.org/.
[3] Kang, B. Hoon and Wilensky, R. Toward A Self-administering Data Model. (forthcoming)
[4] Kahn, Robert and Wilensky, Robert. A Framework for Distributed Digital Object Services. Kahn. Corporation for National Research Initiatives technical report cnri.dlib/tn95-01, May 13, 1995.
[5] The Los Alamos preprint server, http://xxx.lanl.gov/.
[6] NCSTRL, http://cs-tr.cs.cornell.edu/.
[7] Phelps, Thomas A. and Wilensky, Robert. Multivalent Documents: Anywhere, Anytime, Any Type, Every Way User-Improvable Digital Documents. Communications of the ACM, Vol. 43, no. 6, June 2000.
[8] Phelps, Thomas A. and Wilensky, Robert. Robust Hyperlinks: Cheap, Everywhere, Now. In the Proceedings of Digital Documents and Electronic Publishing (DDEP00), Munich, Germany, 13-15 September 2000.
[9] Riggs, T. and Wilensky, R. An Algorithm for Automated Rating of Reviewers. (forthcoming)
[10] Scansoft, http://www.scansoft.com/.
[11] TileBars, http://elib.cs.berkeley.edu/tilebars/about.html#about.
[12] Varian, Hal R. Future of Electronic Journals. In Technology and Scholarly Communication, Richard Eckman and Richard E. Quandt (eds.), University of California Press, 1999.

[13] Wilensky, Robert. Composite Documents. (Forthcoming)
[14] The UC Berkeley Digital Library Project, http://elib.cs.berkeley.edu/.