Multi-institutional EAD: The University of Virginia's Role in the American Heritage Project

DAVID SEAMAN

Abstract: In 1996 the National Endowment for the Humanities awarded a joint grant to the University of California at Berkeley, Duke University, Stanford University, and the University of Virginia to produce a body of Encoded Archival Description (EAD) finding aids and to test the assumption that a collection of such guides could function both as a local resource and as a multi-institutional union database. This paper concentrates on the issues of workflow and on-line delivery for the University of Virginia guides, which employ Web forms for data creation, on-line searching, and "on the fly" conversion to HTML.

About the author: *David Seaman is the founding director of the Electronic Text Center at the University of Virginia. This library service, open since August 1992, combines an on-line archives of thousands of SGML texts and digital images with a center housing equipment suitable for the creation and analysis of text. Seaman has taught e-text and Internet courses at the annual summer Rare Book School at the University of Virginia and has a particular interest in the application of computer technologies to special collections and museums. An earlier version of this paper was presented on 29 August 1997 at the annual meeting of the Society of American Archivists held in Chicago.*

THE AMERICAN HERITAGE PROJECT is a consortium funded by the National Endowment for the Humanities and consists of project teams at the University of California at Berkeley, Duke University, Stanford University, and the University of Virginia. The goals of the consortium are:

- To produce a large number of EAD-encoded finding aids documenting American history and culture;
- To work collaboratively to ensure that these finding aids coexist effectively as part of a multi-institutional "union database;" and
- To examine and report on the intellectual, political, technical, and economic issues that surround the creation of EAD guides.

More information about the project and its participants can be found at the project's website.[1] This article will focus on the University of Virginia's portion of the American Heritage Project.[2]

The NEH grant provided the University of Virginia's Special Collections Department with a full-time staff member and several graduate student assistants for one year to work exclusively on the conversion to EAD of guides that already existed in electronic form (mostly in WordPerfect format). The Special Collections Department handles the selection and conversion of the guides, and the Electronic Text Center provides SGML training and the on-line search-and-delivery tools, thereby effectively combining our various skills and drawing on lessons learned in the past five years of full-text SGML and archival image production at the University of Virginia.

Since 1992, the Electronic Text Center has been serving SGML data on-line, using the OpenText search engine in conjunction with our locally written interface forms and HTML filtering programs. This ongoing activity, mostly focused to date on the use of the Text Encoding Initiative (TEI) Guidelines, has provided a firm basis for the work we are now doing to manage and deliver EAD-encoded data.[3] There are points of similarity between EAD and TEI, especially in the formation of the <eadheader> and in the cross-referencing mechanisms. These similarities helped us deploy EAD rapidly and to convert the TEI authoring and delivery tools to EAD tools.[4]

Methodology

In October 1996 representatives from the four American Heritage institutions met in Berkeley for an intensive series of workshops out of which emerged an "acceptable range of uniform practice" for us to follow in our EAD encoding. This meeting was vital to focus and coordinate the work that we all have gone on to produce. Even for a single-institution EAD project, there is much to be gained by spending significant time examining and discussing with colleagues the level of detail one wishes to achieve in tagging and in the details of production.

[1] <http://sunsite.berkeley.edu/amher>.

[2] The University of Virginia's EAD guides can be found at <http://www.lib.virginia.edu/speccol/ead/>.

[3] For examples of full-text databases at the University of Virginia's Electronic Text Center, see <http://etext.lib.virginia.edu/uvaonline.html>.

[4] The conversion of the TEI Web forms to EAD was performed by Susan Munson, the Electronic Text Center's senior programmer/analyst.

Our series of workshops led to the preparation of an "acceptable range of uniform practice" document, produced by Daniel Pitti for the American Heritage consortium[5] and, for us, to the preparation of a local workflow document for the Virginia data converters to follow.[6] Writing this local document required us to codify—and in doing so, to examine and justify—day-to-day workflow practices in a variety of areas, including:

- file naming,
- file management,
- Unix utilities,
- level and nature of our tagging, and
- indexing and browsing.

Despite being project-specific, we hope these "acceptable range of uniform practice" and workflow documents may serve as useful starting points for other EAD projects.

The formulation of a written manual is highly recommended on such a project for a number of practical reasons, not least of which is project continuity when a key member leaves for another job. This happened to us five months into the project, and the "corporate memory" of the project that the workflow document encapsulates greatly eased the transition period for a new full-time EAD staffer. Our experience repeatedly has been that the time spent building such a "project manual" is one of the best investments one can make in a new project.

The current finding aids at the University of Virginia library exist variously in paper format, plain ASCII text, and WordPerfect. Fortunately, we have enough guides in electronic format to cover most of our production needs for the American Heritage Project, and so we are converting few guides from paper as part of this project. In the year since the project began, we have set up the operation, trained the staff, coordinated with the other institutions, and converted over one thousand guides, including parsing and proofing the results after they appear on the Web.

Conversion

We employ two methods to create and convert our EAD guides: 1) For the EAD header and front matter, we use fill-in-the-blank Web forms, and 2) for the body of the text (main scope and content summaries, and the item-by-item summaries), we use WordPerfect conversion techniques.

1. Web forms for the EAD header and front matter. The use of Web forms for standard sections of an SGML document such as the "metadata" header (the <eadheader> element in EAD) has been a practice for several years at the University of Virginia (UVa), in particular for the TEI header we complete for each electronic text that we build. The system we use is a locally produced program, written and distributed by the Institute for Advanced Technology in the Humanities, and adopted gratefully by the Electronic Text Center. In return, we plan to spend some of the center's programming time in late 1997 to augment and develop this program to perform the following additional functions:

[5]See <http://sunsite.berkeley.edu/amher/upguide.html> for the EAD "Retrospective Conversion Guidelines" intended to represent an "acceptable range of uniform practice."

[6]The Virginia guide to EAD conversion can be found at <http://etext.lib.virginia.edu/ead/eadguide.html>.

- Radio buttons and/or pull-down menus on the Web form for boilerplate text; and
- The ability to enter both text and tags into the Web form fields (at present one can enter only text). For simple documents, it may well be possible to use the form to enter not only the header and front matter, but also the Description of Subordinate Components <dsc> information.

Even as it stands, the Web form is a very useful tool. The program reads an SGML template file and configures itself accordingly, providing fill-in boxes for some or all of the tagged fields in the template. If the template file contains boilerplate language for certain fields, that material also appears on the form. Therefore, it is not a "hard-coded" form that knows only one set of fields, which means that we can use it for other projects simply by providing it with another set-up template. Moreover, one does not need to complete a form at one sitting; an encoder can come back the next day and pull the half-completed data back into the Web form, or call up a completed draft to make revisions. The completed form is saved directly to the Web server. (For the Text Encoding Initiative headers we create for our electronic texts, the Web form also automatically creates a MARC record from the SGML file, and this may be possible for the EAD guides as well.)

The use of such Web forms for data creation has three major advantages:

- An encoder cannot inadvertently create invalid tags, because the tags are hidden from the user and supplied by the form;
- The documents created through the forms have an absolutely consistent similarity of structure, which makes them easier to change programmatically at a later date; and
- The data is saved directly to a directory on the Unix Web server, which has higher-level data security backup mechanisms in place than we typically have on our desktop computers, and which also makes file sharing easier.

2. WordPerfect macros. Predictably, the WordPerfect "legacy data" finding aid files vary somewhat in their coding and layout, but they are similar enough in design to be converted to EAD-tagged data by a combination of WordPerfect macros, search-and-replace operations, and manual tidying up. This process is eased by the fact that our paper guides are rarely tabular in their layout, but rather, are organized in a list format. While EAD provides detailed support for tabular layout, the tagging is complex and is currently difficult to deliver on-line, because both Web browsers and the current popular SGML viewer, Panorama, have difficulty dealing with large tabular documents. I suspect that even if our guides were in tabular layout on paper, we would think seriously about the desirability of continuing that layout into the electronic versions.

We are not using an SGML editor for retrospective conversion. While either WordPerfect 7 and 8 or Author/Editor would be good candidates for authoring new documents (and it is likely that we will use one or both of them for that purpose in the future), these editors are not document-conversion tools; they are at their best when one is writing a new finding aid and encoding it in EAD from the outset.

On-Line Delivery

As with the Web data-entry forms, the search interface, HTML "on the fly" filter, and the Table of Contents generator described here all are items that we have developed over the past five years for handling our full-text and image databases. We have been very pleased to discover that the heavy investment that the Electronic Text Center has made in

these software tools continues to transfer to new projects and new SGML Document Type Definitions (DTDs).

1. Searching. In order to search the growing archives of EAD guides, one simply fills in a Web search form that allows the user to search for words, phrases, and items in some proximity to one another, and then to limit further by EAD tags or collection names.[7] For example, one may look for "Chicago" everywhere in the collection of finding aids or limit the search term only to the UVa Mark Twain collection (see Figure 1). The Web form passes the query back to the OpenText search engine, which executes the search; the results come back from the search engine and are piped through an HTML filter and sent out to the user.

What one sees first is a Keyword-in-Context (KWIC) display—the search results with a small amount of surrounding context and the name of the collection from which the results came. The user can then choose to go on to larger and larger contexts around the search term: the section in which the term appears, the entire description of items, or the entire guide. In this fashion we make use of the hierarchical nature of SGML to display the search term in different contexts.

2. Browsing. An alternative way of accessing the collection of finding aids is to go to a browseable list of titles and simply select one.[8] A program written at the University of Virginia builds these pages of links dynamically, drawing on SGML fields in the EAD-encoded data. When we add a new guide, we rerun this script, and the pages are updated. The script pulls out titles, which it arranges alphabetically by surname, with each title followed by accession numbers and file size.

Here again the conversion of the EAD tags to HTML is achieved "on the fly" as it is displayed to a user. This means that we do not need to maintain a static HTML copy of a guide on the server in addition to the SGML version, thus avoiding all the attendant problems of tracking and updating two copies of a file.

3. Automatic Table of Contents Generation. As an alternative to retrieving the entire file, one can use a Table of Contents generator—the "TOC" choice displayed at the end of each entry in the lower left corner of Figure 2. This program takes advantage of SGML's predilection for hierarchical nested structures and predictable title and caption fields: it breaks out "on the fly" the hierarchical structure that exists in the EAD guide and builds a table of contents page that allows one to choose to view only a portion of the file—for example, the front matter or a <c01> or <c02> section of the container list. As a navigational aid, the TOC program prints out for each choice the <head> or <unittitle> that belongs to it (an example can be seen in Figure 3). The ability to browse only a part of the file really comes into its own with a large file such as the Mark Twain Collection.

4. HTML conversion "on the fly." Whether one chooses to search across our collections or to browse a single collection (or some piece of it from the Table of Contents generator), the results are delivered through the same EAD-to-HTML filter, giving the results a uniformity of appearance. The results of a browse or a search are piped through a perl script that substitutes the EAD tags for HTML tags. A simple example follows:

[7]To try this, see the "Search UVa Guides" choice at <http://www.lib.virginia.edu/speccol/ead/>.

[8]To try this, see the "Browse UVa Guides by Name/Title" choice at <http://www.lib.virginia.edu/speccoll/
ead/>.

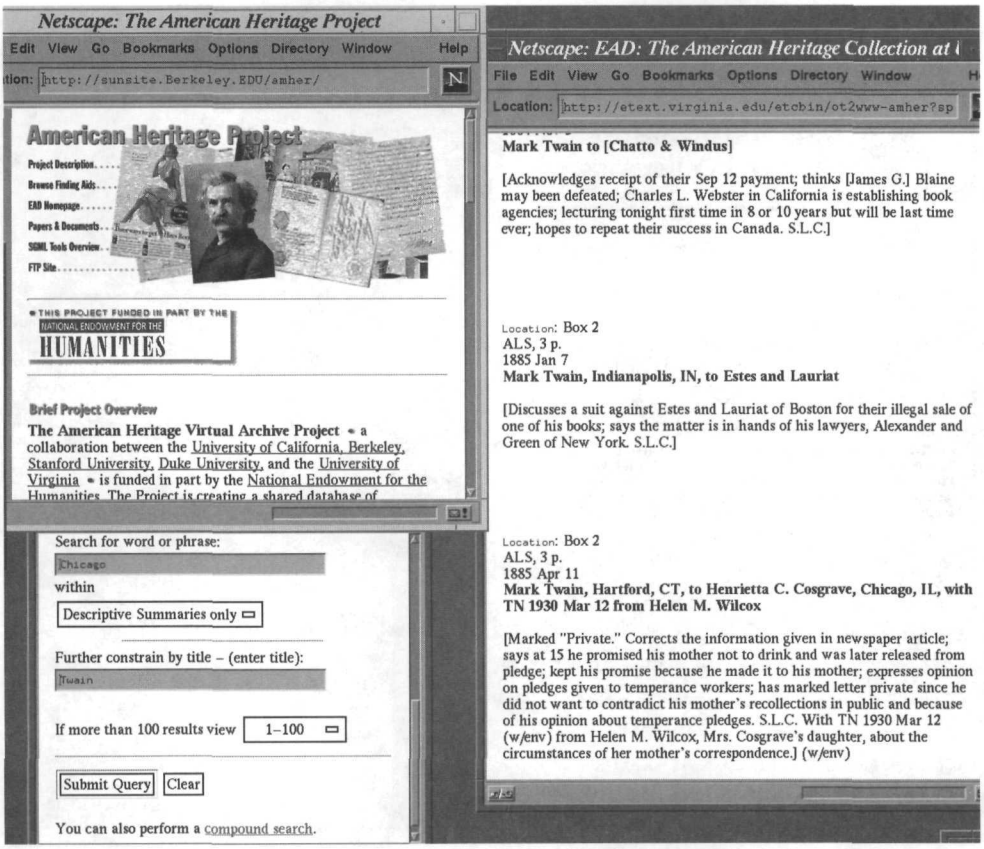

Figure 1. A search for the words "Chicago" and "Twain," each within specific finding aid segments.

The perl script

```
s/ <unittitle> / <center><h2> /g;
s/ <\/unittitle> / <\/h2><\/center> /g;
```

would convert the EAD tags

```
<unittitle> Manuscripts </unittitle>
```

to the HTML tags

```
<center><h2> Manuscripts </h2></center>.
```

An obvious advantage of this is that we can make global changes to the appearance of our documents simply by changing a line or two of instructions in the filter.

We do not rely on SGML helper applications such as Panorama because they do not solve any particular issues for us, and, in fact, they create some problems:

- Users resent having to set up another piece of software on their client machines simply to look at a single finding aid;
- Unix, Mac, and VT100 users are excluded, as no version of Panorama exists for these platforms; and

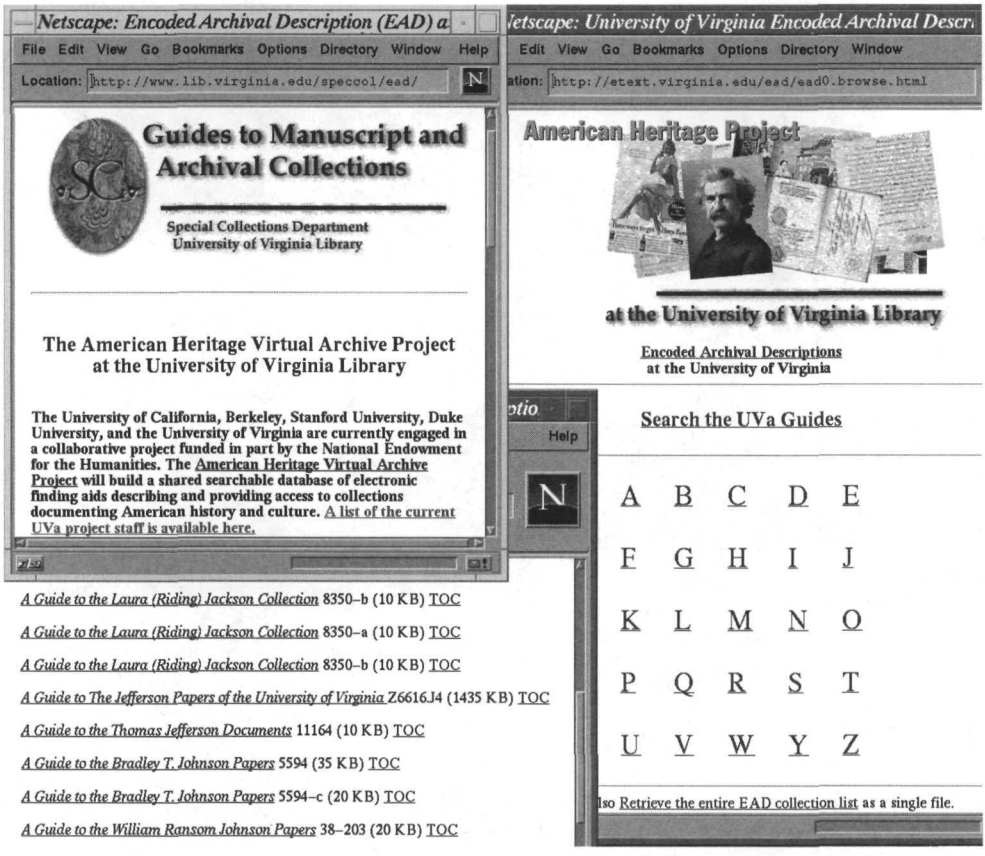

Figure 2. Use of the table of contents generator.

- Users on slower connections resent being forced to download an entire document instead of simply the piece of the document—a <c01> for example—that they need. The SGML helper applications do not currently allow one to request a portion of a longer document.

Looking Ahead

The UVa Special Collections are using the current retrospective conversion project to think through very clearly what the shape, format, and contents of future guides created directly in EAD should be, now that we have a searchable, browseable electronic environment in which to place them rather than simply a paper medium. This initial period of EAD creation allows reflection on what the sensible, functional level of markup should be—what the tagging "sweet spot" is—and allows us to experiment with guides that combine EAD summaries with links to digital full-text and image versions of the objects they describe. A good example would be the James and John Booker Civil War letters.[9] The combination of full-text and image items that are available on-line for this collection

[9]See <http://etext.lib.virginia.edu/ead/eadB.browse.html>.

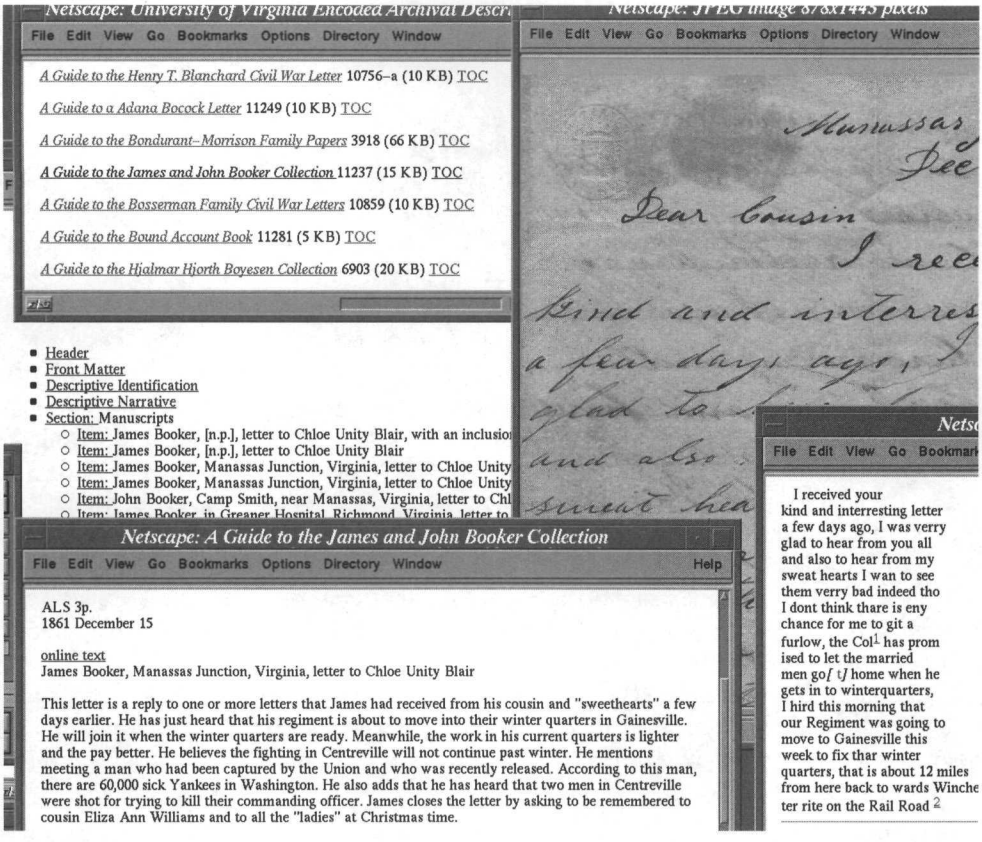

Figure 3. Search results (full-text and image items) demonstrating use of the table of contents generator.

adds to the level of excitement and makes the value of archival description much more evident to nonarchival users (see Figure 3).

We also keep reminding ourselves of the promise of a national union database of EAD guides, and the incredible research potential that would be enabled by the ability to perform high-level national or regional searches. The multi-institutional American Heritage Project is testing this potential in a very practical manner: each of the four institutions delivers finding aids regularly via FTP transfer to Berkeley, where they are being assembled into one large searchable and browseable database.

Conclusion

In conclusion, I offer four suggestions to archivists contemplating implementation of SGML-encoded finding aid systems.

1. Partnerships. Much of what we do at the University of Virginia relies on good working relationships with others in the organization who have skills necessary to the success of our digital library projects, including catalogers, collection development specialists, archivists, and systems specialists. Archivists working in smaller institutions may have to go outside their immediate environment for these partnerships, perhaps by forming a consortium with neighboring institutions.

2. Take advantage of workshops. Working in isolation is ineffective, and one should try to take advantage of conferences, workshops, and professional meetings to learn new skills and share discoveries prior to attempting implementation of a technology such as EAD.

3. Set achievable goals. Take on small projects initially; for example, ones that can be brought to completion in a semester. Such an approach provides the benefits of a quick result for demonstration purposes, as one tries to raise user awareness and interest while pursuing local and national funding opportunities. Such small projects also can be useful for working out local data creation or conversion processes.

4. Remember that SGML is your friend. While learning and implementing EAD is not a trivial undertaking, it is a manageable one, as is witnessed by the rapidly growing numbers of institutions taking on EAD projects. The investment in EAD over plain ASCII text or HTML is repaid quickly as one starts searching and delivering the guides, notwithstanding the current paucity of good, cheap SGML tools. With WordPerfect's support of SGML, and with important advances such as Extensible Markup Language (XML) on the horizon, there is a firm sense that the use of SGML DTDs other than HTML is entering into the mainstream, precisely because of the longevity and explicit nonproprietary data structure that have led to SGML's growing adoption by libraries and publishers over the past decade. An EAD guide may not have any more tags in it than does an HTML version, but the tags are descriptive of the nature and structure of the data rather than simply describing its appearance.

In all of this, it is salutary to remind oneself that we are still working in something of a "frontier culture" (the work reminds you of this with some frequency if you forget), not in a settled environment, and there will be moments of frustration along with the warm glow of EAD success.