Over, Under, Around, and Through:
Getting Around Barriers to EAD Implementation

Michele Combs
Syracuse University

Mark A. Matienzo
Yale University

Merrilee Proffitt
OCLC Research

Lisa Spiro
Rice University

Over, Under, Around, and Through: Getting Around Barriers to EAD Implementation
Combs, et al., for OCLC Research and the RLG Partnership

Suggested citation:
Combs, Michele, Mark A. Matienzo, Merrilee Proffitt, and Lisa Spiro. 2010. Over, Under, Around, and Through: Getting Around Barriers to EAD Implementation. Report produced by OCLC Research in support of the RLG Partnership. Published online at: www.oclc.org/research/publications/library/2010/2010-04.pdf.

Contents

Introduction

This report frames some of the obstacles that archivists have experienced adopting Encoded Archival Description (EAD). It also suggests pathways to help you get out of the ruts, around the roadblocks, and on the road to success. This report is addressed to those who have a basic understanding of standard archival descriptive structures and modest acquaintance with EAD. Our objective is to help you communicate EAD's value as a key element of successful archival information systems and overcome potential barriers to its implementation. This paper does not contain an EAD primer, or cover the basics of document encoding. For those who are not familiar with EAD, we recommend the EAD Help Pages as an excellent starting place for more information.[1]

Archivists have been encoding finding aids using EAD for over a decade. An impressive number of institutions have implemented EAD, but many have not. A 2008 survey revealed that nearly half of respondents (79 out of 168) had not yet implemented EAD.[2] A further analysis of the characteristics of those who had not yet adopted EAD reveals that all types of institutions are represented, including archives affiliated with large and small universities and those with a range of information technology (IT) services, from no professional IT staff to those with access to the services of a large IT department.

Our professional literature articulates obstacles ranging from political to technical, and much in between. over the last ten years a growing body of relevant articles detail barriers: Jill Tatem's article "EAD: Obstacles to Implementation, Opportunities for Understanding"; James M. Roth's "Serving up EAD: An Exploratory Study on the Deployment and Utilization of Encoded Archival Description Finding Aids"; and Elizabeth H. Dow's "EAD and the Small Repository."[3] These early works were followed by Katherine M. Wisser's EAD Tools Survey and Sonia Yaco's article, "It's Complicated: Barriers to EAD Implementation."[4]

Political or logistical issues may keep you from getting going; technical issues may get you bogged down along the way. Against this backdrop of challenges, there are a growing number of tools that support EAD.[5] Nevertheless, real and perceived barriers to EAD implementation still exist, all of them well documented. For every roadblock, as Sesame Street's Grover says, there is a way "over, under,

around, and through."[6] This paper presents useful tools—informational, persuasive, or technical—for overcoming barriers you may encounter in your journey towards EAD implementation.

Section I of this report addresses political and logistical issues. These include gaining buy-in from institutional decision makers, overcoming the urge to achieve perfection, finding ways to maximize scarce resources, and getting over the initial humps of dealing with a relatively complicated standard and what can be perceived as overwhelming logistical issues.

Section II navigates technical problems and solutions, such as thinking about lossless data streams in conversion and management, selecting software (and challenges around open source software in particular), publishing, and mitigating the complexity of the standard.

Members of this working group (under the auspices of the RLG Partnership and OCLC Research) authored this report jointly. We all have had experience with EAD and have struggled with the range of issues. Thus, the advice we offer comes from practical experience.

This paper addresses a wide range of needs because of the assortment of issues. We hope that you will dip directly into the sections that are most appropriate to your particular need. We present barriers as articulated in published literature. We then propose one or more solutions that may work for you. Our goal is to show you that implementing EAD is easier than you think. We hope these strategies will be helpful and will smooth the way to successful implementation.

Section I: Political and Logistical Issues

I'm preaching to the unconverted[7]

Solution: Prepare effective arguments about EAD's significance

The following arguments may help you communicate that EAD is a good investment of institutional funds and staff resources. We begin with a brief "elevator speech" to introduce the nature and purpose of EAD, followed by more specific points.

The elevator speech—What is EAD, and why should my institution invest in it?

EAD is an international standard for encoding finding aids established to meet the needs of both end-users and archivists. EAD is represented in XML (Extensible Markup Language), a platform-neutral data format that ensures data longevity when migrated from one software environment to another. EAD ensures the long-term viability of your data by encoding intellectual rather than only presentational data (HTML, for example, only accomplishes the latter). EAD can be produced from (or mapped to) a variety of formats, including relational databases, MARC, Dublin Core, HTML and others, which makes it an excellent format for porting data. In addition researchers can have a more robust interaction with EAD finding aids because EAD enables better searching and subsequent delivery from a single source document.

...and more!

Pick and choose from among the following ideas that will be the most persuasive in your circumstances.

EAD is an internationally-used encoding standard

EAD complies with data content standards such as ISAD-G (the *General International Standard Archival Description*, developed by the International Council of Archives) and DACS (*Describing Archives: A Content Standard*, developed by an international working group under the auspices of the Society of American Archivists).[8] EAD is global; EAD has been implemented by a wide variety of institutions, not only in the US and Canada, but also throughout Europe, Australia, New Zealand and Asia.

EAD plays well with others

EAD has been mapped to and from other data encoding standards such as MARC and Dublin Core.[9] Because EAD supports hierarchical description, you can map data from a relational database; many commonly-used EAD tools are, in fact, built on relational databases. EAD need not be the environment in which you produce, store and manage your description, but it works well as a global transfer syntax.

EAD encoding facilitates aggregation

It would be difficult, if not impossible, to create effective subject gateways like the American Institute of Physics' Physics History Finding Aids Web site, or regional collection gateways such as the Online Archive of California, without the consistency imposed by EAD.

An abundance of tools support EAD implementation

Tools exist to facilitate every aspect of EAD use, from encoding to publication. So many tools exist that we've included only a selection in Appendix II. An even wider variety of tools are covered in *Archival Management Software: A Report for the Council on Library and Information Resources* (2009).[10]

EAD implementation is supported by significant opportunities for training and collaboration

Opportunities abound for formal and informal EAD training, advice and consultation to support the growing population of EAD implementers. Some examples include the EAD discussion list, courses offered by the Society of American Archivists and Rare Book School, and workshops at local, regional and national conferences.[11] Various state and regional consortia offer EAD training opportunities, tools, and guidelines.

EAD is good for researchers

...in a number of ways:

1) Researchers can discover collections in more places through wider availability. EAD's consistent coding and structure means it's easy to submit your finding aids to multiple access points (to the Online Archive of California, or to OCLC's ArchiveGrid, or to a subject-based portal such as the one maintained by the Niels Bohr Library & Archives at the American Institute of Physics, for example) so they're more likely to be found by researchers.

2) Inexperienced researchers can use finding aids more easily. Consistency of content and presentation eases the use of collection descriptions for inexperienced researchers. Finding aids that are exposed online are far more likely to be found by inexperienced researchers—

an audience whose needs we must always bear in mind—than collection descriptions that are only available locally.[12] As user studies reveal better and more intuitive ways to present finding aid content, reformatting collection guides encoded in EAD is painless. If one presentation/display method proves problematic or confusing for researchers, you can change it with minimal time and effort and zero rekeying or editing.

3) Researchers can filter and refine searches. Some applications can utilize EAD's structured tags. This makes it possible to limit searches to scope and content notes or collection titles, for example.

4) Display and output can be tailored for research needs. One single EAD encoded file can provide multiple output versions for multiple researcher needs (online version, printer-friendly version, etc.). You can also easily create different display options for different audiences.

5) Researchers can explore old data in new ways. EAD enables archives to offer researchers new, interesting, powerful, and productive visual explorations of collections. There are some great new tools under development. Examples include: Jeanne Kramer-Smyth's ArchivesZ, an "elastic list" prototype at Syracuse, and relationship mapping tools such as NNDB Mapper.[13]

EAD gets you money

Grant agencies and other funders look favorably on and encourage EAD implementation as part of their granting process. For example, the guidelines for the NEH Preservation and Access, Humanities Collections and Resources encourage the use of EAD.[14] The NISO/IMLS A Framework of Guidance for Building Good Digital Collections includes EAD as an appropriate metadata scheme for archives.[15] NHPRC similarly endorses EAD in their guidelines.[16]

Knowledge gained mastering EAD is applicable in other contexts

In learning EAD, you will also develop skills that extend beyond encoding finding aids by gaining a basic understanding of XML and XML tools. So much digital data—in the library and archival communities and beyond—is stored and/or exchanged in the form of XML. These skills for staff will allow them to work with other standards such as MARCXML, MODS, and METS.

EAD paves the path to the future

Although today's researchers find collection descriptions using keyword searching on search engines, the Web of the future will be no place for unstructured data. The future is the "semantic Web" or linked data. Implementing EAD will help to position your institution for the future of internet applications.

Everybody's doing it!

Recognition of EAD's significance has become increasingly widespread, both within the US and internationally, and its use has expanded accordingly. The EAD Help Pages include a comprehensive list of various types of institutions that are currently successfully implementing EAD. [17] We encourage you to look for institutions with a profile similar to yours. Knowing that your peer institutions are implementing may help you persuade those that hold the purse strings that EAD implementation is worth the resources it will take.

Let me just tidy this up first...[18]

Solution: Encode the data you have to provide minimum access.

As Voltaire wisely observed, "the perfect is the enemy of the good."[19] The desire to achieve perfection can sometimes get in the way of small improvements that iteratively help us to reach larger goals. In the case of providing better access to our collections, the urge to rewrite finding aids (or reprocess collections and then rewrite finding aids) is a huge barrier to providing interim access to the collection descriptions as they are now.

You must make every effort to make existing collection descriptions as accessible as possible, regardless of your intentions for them in the future. Although technically these collections do not represent a processing backlog since they do have descriptions, if those descriptions are not accessible, they present the same problems as unprocessed collections. They are hidden from all but those inside the institution. A survey conducted in 2003-2004 by Dennis Meissner and Mark Greene as background for the "More Product, Less Process" report found that backlogs are a key concern for the majority of donors, researchers, and resource allocators.[20] SAA's code of ethics reminds us that "Archivists strive to promote open and equitable access to their services and the records in their care without discrimination or preferential treatment."[21] Likewise, the SAA/ALA Joint Statement on Access includes the following statements: "A repository is committed to preserving manuscript and archival materials and to making them available for research as soon as possible" and "As the accessibility of material depends on knowing of its existence, it is the repositories responsibility to inform researchers of the collections in its custody."[22]

In an increasingly online world, making your collection descriptions as accessible as possible to the widest possible audience is of paramount importance. Access deferred is access denied.

Who will do the work, and when?[23]

Solution: Find low-impact ways to tackle EAD implementation .

Your staff is already stretched to the limit, and adding EAD implementation to an already bulging workload is rarely feasible. Taking a creative approach may help your institution get a toe in the door with a modest investment of time or money.

Join the club

Numerous state-, regional- or subject-based consortia have pooled resources to benefit member institutions by lowering barriers to EAD implementation. They generally offer a range of services such as best practice guidelines, stylesheets, templates and other tools, training, and hosting of data. They often apply for grant funding to convert finding aids or provide other services at little or no cost. Some consortia may be able to handle all of your needs, including encoding. Even if you don't belong to a consortium, many of these organizations make their tools and guidelines freely available on their Web sites, so that others need not reinvent the wheel. Many consortial projects are grant-funded; the more contributors and users they have, the more likely they are to continue being funded. Contributing your EAD thus helps not only you but many other institutions and patrons. A partial list of regional, national and international consortia is included in Appendix I.

Take the first step

If you have collection-level records in the MARC format, consider creating basic EAD records through export (easily done using a tool such as MARCEdit).[24] You will then have a set of collection-level EAD data, and some experience working with the standard. The resulting files are also suitable to contribute to an EAD consortium. You may then decide to expand the minimal records, or you may decide to live with the fact that your EAD descriptions will not include inventories. Something is better than nothing, particularly from the user perspective.

Take an iterative approach

There is no rule that EAD encoding must be done once and only once, nor that it cannot be done until arrangement and description are complete. Implement EAD with a "More Product, Less Process"-like approach! A collection-level finding aid with minimal information can be produced at the time of accession. Areas such as `bioghist` and `scopecontent` can be expanded later. If the collection comes with a simple box list, that can be included at the outset, to be replaced later by a fuller inventory when processing is complete. Some collection management tools will produce collection descriptions as a byproduct of the arrangement and description process. EAD finding aids and online publishing free us from the static paper finding aid, offering instead an evolving document that changes and grows though the life of the collection.

Use someone else's time and talent

Do you have access to library school students, or other interns? Do you have an internship program? Consider offering an internship (or a series of internships) that focus on researching options and implementing EAD for your institution.

You don't have the time, but you do have the money

Maybe you don't have staff time, but you can make a one-time or ongoing investment of funds. If wholly or partially outsourcing is of interest, there are more details on page 19.

It's so complicated![25]

Solution: There are many options that make EAD simple

EAD can intimidate even tech-savvy staff, given the number of tags and the seemingly endless variety of ways they can be implemented, but not all elements are required. Use collection-level descriptions and minimum-level description elements—as given by DACS—to simplify EAD adoption.

A variety of tools exist to help mask the complexity of EAD and smooth the encoding path. Tools that assist in migration can also aid in the tagging process. See the section on migrating to a database or content management system (page 21). Other tools that can help mask complexity are listed in Appendix II. Please note that some of these tools are local solutions or strategies—that is to say, they have been customized for use within a consortium, institution or repository and may not work in your setting. We've included these because they may provide inspiration for your own use. The section on "Specialized Migration and Conversion Tools" may be especially useful (page 36).

Templates

Templates are EAD documents pre-populated with text that never changes (repository name, address, etc.) and with boilerplate text guiding the encoder to fill in proper data. This removes much of the angst of choosing what tags to use and how to use them, making it more like filling out a form.

Templates are simple to create. Using commercial XML authoring tools, you can create an EAD file with as much information as possible—including both text and attributes—already filled in, and use it as the basis for all new EAD files. For example, in the `controlaccess` section include one each of the possible child elements (`subject`, `persname`, `corpname`, etc) with the appropriate `@source` and `@encodinganalog` attributes filled in. Or in the `bioghist` section, include the phrase "MARC 545: Insert brief bio or company history." Some example of templates are included in Appendix II.

The EAD Schema

Using the schema rather than the DTD version of EAD during authoring allows you to enforce various content limitations, such as correct formatting of `@normal` attributes for date elements, thus reducing the chance for errors. While leveraging the full power of the EAD schema may require a more substantial investment of time, schema-based validation can be used in combination with templates to ensure strict internal compliance. An example of institution-wide best practice guidelines implemented using the XML Schema version of EAD is that of Yale University's Finding Aid Coordinating Committee. Yale finding aids validate against the W3C Schema version of EAD, and compliance with Yale's EAD best practices is monitored via external validation against a RelaxNG schema.[26]

Forms

An EAD finding aid can be presented as a fill-in-the-blank HTML form so that the archivist never sees any EAD at all. Although it does require behind-the-scenes coding to add the EAD tags, several institutions have employed this approach and there is likely a Web-based form that your institution can use immediately, or with minimal adaptations. We've included examples of several Web-based forms along with two examples of stand-alone forms in Appendix II.

Style sheets

Style sheets can display or hide various attributes or other text to enhance readability and aid in completing a template. If using oXygen, XMetaL, or some other commercial XML authoring tool, a Cascading Style Sheet (CSS) file can be automatically created the first time you open an EAD document; this style sheet can then be modified, enhanced, etc., to assist editors visually. Some examples of stylesheets are included in Appendix II.

...and more!

We've included information about other means of simplifying EAD implementation including pointers to commercial XML tools, content management systems for archives, and a variety of papers, production guides and case studies. You will find lists in Appendix II.

I don't know where to start![27]

Solution: Take it one step at a time and create a plan. The process can be broken into logical steps.

If you are overwhelmed and don't know how to get started, it may be helpful to think about EAD implementation in terms of a number of small steps. Start with what you have and where you need to go. Steps usually include documenting inhouse standards (e.g., should extent be expressed as linear feet or cubic feet? Are there existing best practice guidelines you wish you adhere to? What controlled vocabulary will you use for the `controlaccess` terms), selecting an encoding method (e.g. template with XML editor, database, full content management tool), data entry, selecting or creating a style sheet (to format the XML), and putting the files online. Additional decision areas may include whether and how to provide search capability, whether to provide alternate formats (e.g., a printer-friendly version), whether to link finding aids to digitized content, and so on. For an overview of possible steps, we have included one such plan in Appendix III, and this example may get you started. You may also refer to the *EAD Application Guidelines*, specifically Chapter 2, for more details and additional food for thought.[28]

Section II: Addressing Technical Problems

I want my data to be stored in a format that will give me flexibility going forward[29]

Solution: EAD is a non proprietary component of a flexible framework.

In any conversion process, you should be concerned with maintaining flexibility, and not losing information. Converting paper files to electronic format increases the usability of your data; EAD encoding offers both flexibility and additional options since EAD can be "crosswalked" into other standards like MARC XML and Dublin Core. Going further, a well-chosen database or CMS (Content Management System) can provide additional output options. It's all about choices along this continuum. We've presented three ways to think about moving forward—each one will advance you to greater flexibility.

Option 1: Migrating hard copy or word-processing files to EAD

Outsource

Outsourcing data conversion is the fastest and easiest solution for hard copy or word-processing file conversion. Depending on your staff costs, this may or may not be the cheapest option. Generally conversion houses quote a per-page (for hard copy) or a per-byte (for electronic files) rate. Outsourcing requires considerable up-front work in determining tagging specifications, since EAD is highly flexible. Adhering to DACS, ISAD(G), and/or RLG's EAD Best Practice Guidelines[30] or other widely-accepted standards or guidelines is strongly advised. Standards and "best practices" save time and effort, and your end product will be more likely to work with widely-used tools and in aggregations of other EAD files. The EAD Listserv is an excellent resource in terms of getting up to the minute information regarding current vendors.

Outsourcing is most successful when both parties are very clear on the encoding specifications. A sample encoding specification, including where to get various pieces of data and how to handle particular situations is available on the Syracuse University Library Web site.[31]

In-house encoding

In-house conversion offers numerous benefits: fostering staff skills, flexibility in schedule and workflow, and direct control over process and inputs. Basic XML skills are not difficult to acquire, and having internal staff with XML knowledge may benefit other library processes and projects as well. Various tools can speed/ease creation of EAD. The two following options—authoring tools and scripts—assume starting with electronic files (e.g., word processing files, or text files obtained through converting hard copy using optical character recognition, or OCR).

As part of in-house conversion you can use commercial XML authoring tools (we've included a list of the most well-established in Appendix II). You may also be able to manipulate the text using scripts. Once you have gained confidence in understanding and defining the EAD output you expect, then any programmer with experience of scripting languages like Perl or VisualBasic and XML could write scripts to produce the desired output from your existing input. EAD produced this way can be validated and/or edited using commercial tools, or using the free online W3C validator[32] or validation could be part of the scripted process(es).

Option 2: Exporting EAD from a content management system

Some content management systems (CMS), such as the Archivists' Toolkit or Archon, are capable of producing EAD. If your CMS does not output EAD directly, two key questions are 1) whether required EAD elements are separate fields in the database, and 2) whether your database exports XML.

Table 1. Tips for producing EAD from managed content under various export scenarios.

Separate fields?	Data Exports as XML?	Notes
Yes	Yes	Export the data as XML and then use XSLT to convert to EAD. (XSLT is a language for transforming XML documents into HTML documents or to other XML documents, in this case transforming data to EAD). This requires some knowledge of XSL, or the funds to contract out XSL development. Ideally it would be a one-time cost.
Yes	No	Export the data in some other structured form (comma-separated values, for example) and identify a scripting approach to process the data and convert to EAD. This requires some programming or scripting knowledge. Perl is an example of a scripting language that is useful in this context.
No	No	Determine whether scripts can be written to parse output from your database and generate EAD; whether, and if so how, the database needs to be modified to be able to export EAD; or whether perhaps the appropriate solution is migration to an EAD-capable database.

Option 3: Migrating to a database or content management system capable of producing EAD for permanent storage and maintenance

Using a database to create and store data elements of finding aids simplifies data entry, reduces the possibility of tagging errors, ensures consistency in output, and offers the possibility of exporting to formats other than EAD. However, some full-fledged archival management systems may be "overkill" for a legacy conversion project in terms of features, price, and learning curve.

If your data is in spreadsheet or word processor format, or in a database that will not map directly to EAD, migrating to an EAD-capable database may be a useful solution. The key question here is whether the data is easily mapped to the target database, and whether the time involved in migration will in the long run result in the best solution for your needs. A list of content management systems is included in Appendix II.

Doors are closed to open source[33]

Solution: Outline the upsides of open source software

Making a choice

The open source/commercial distinction is one of many factors that should play a role in your archival management system decision-making process. The most important part of selecting a system is to choose one that has the features you need. Resources such as *Archival Management Software: A Report for the Council on Library and Information Resources* will help you with the selection process.

Availability of open source software

At least two tools that produce EAD are distributed as open source software (OSS).[34] OSS is produced in a way that allows others to adapt, modify and redistribute the underlying code and is frequently associated with a "community" of developers.

Your institution is most likely using open source software already in some context and that may make it easier to bring in an open source EAD tool. You may be using the Apache Web server, database platforms such as MySQL, and desktop applications such as the Mozilla Firefox Web browser. Several open source integrated library systems are available, including Koha and Evergreen[35] Open source digital repository systems include Fedora, DSpace, EPrints, and Greenstone.[36]

Lack of conflict with commercial software/commercial enterprises

Some institutions have a policy against implementing open source software, preferring instead to license or purchase software that includes support or is backed by a reputable company. Open source software does not preclude commercial support. Support contracts are available for many open source software packages, including the open source ILS system previously mentioned. Commercial support for OSS EAD tools is not currently available, but this is evolving.

Problems with publishing[37]

Solution: Let the browser do the work, or use existing tools that incorporate publication functionality.

A major obstacle preventing wide-scale adoption of EAD is delivering EAD-encoded finding aids to users online. Creating EAD finding aids may require a different set of skills than publishing them, including authoring XSLT stylesheets, installing software, configuring a server, and so forth. There are few inexpensive, "out-of-the-box" solutions for publishing EAD online. However, archives have several options. From simplest to hardest, these include: contributing records to a shared finding aids repository; delivering EAD directly to the browser; converting records to HTML or PDF for Web display; using inexpensive tools to enable searching of HTML and XML files; using an archival management system; and using an XML publishing platform.

Contribute to a shared finding aids repository

Rather than developing their own technical infrastructure for delivering finding aids, some archives choose to deposit them in regional finding aid repositories. The finding aids are hosted centrally and provide a single point of access to finding aids from multiple institutions. We've included a partial list of finding aid repositories/regional consortia in Appendix I.

Some archives may want to contribute finding aids to a repository *and* make them available via their own Web sites.

Deliver EAD directly to the browser

This is by far the simplest and easiest approach. You can deliver XML directly to most recent Web browsers (e.g. IE 5+, Firefox .9+). To transform the EAD XML file to HTML within the Web browser (on the client side), include a processing instruction in the XML document pointing either to an XSLT stylesheet[38] (the preferred method) or CSS file.[39] However, some institutions may not want to provide access to their raw XML files, particularly if they include sensitive information in their finding aids that they don't want to display to the public. Moreover, browser support for XML is still

uneven[40] (for instance, at the time of the writing of this report, Google's Chrome browser is reported to not provide full XML support).

Convert your EAD to HTML or PDFs for Web display

Instead of displaying the raw XML using a Web browser, convert EAD finding aids to a static files in a human-readable format. By applying XSLT stylesheets to XML finding aids, archives can generate multiple forms of output, including HTML and PDF. Such conversion can be accomplished in batch. HTML or PDF files can then be uploaded to a standard Web server to support research and discovery.

Developing XSLT stylesheets requires some technical knowledge, but several consortia and archives have made available XSLT stylesheets that archives can easily adapt for their own institutions. Some examples are listed in Appendix II.

Delivering HTML or PDF rather than EAD may be attractive to archives that lack technical staff to support XML publishing, but these methods have several drawbacks. They do not take full advantage of having archival information marked up in EAD; searches cannot be restricted to particular EAD elements. Moreover, every time the finding aid is updated, the HTML must be regenerated and uploaded to the server. Some archives use a hybrid approach; indexes are created from EAD files to enable fine-grained searching, but the HTML file is delivered to the user when they want to view the finding aid. Syracuse University Libraries take this approach.[41]

Use inexpensive tools to enable searching of HTML and XML files

Even if an archive lacks a substantial budget or large technical staff, it can choose from several inexpensive, easy-to-implement tools that support indexing and searching EAD files. One example is Swish-e, "a fast, flexible, and free open source system for indexing collections of Web pages or other files."[42] Google Site Search also provides an inexpensive, customizable way of searching your Web pages.[43]

Use an archival management system that supports publication

Many archival management systems enable publication via export of finding aids in EAD, HTML, or PDF. By using archival management systems, archivists can streamline workflows, avoid duplicating data in multiple places, find and share information more easily, manage collections, and generate reports and statistics.[44] A list of archival management systems that support Web publishing of finding aids are listed in Appendix II.

Archival management systems have some drawbacks: they may enforce a rigid workflow, it can be difficult to import data, and some are costly to implement. On the other hand, archival management systems can enable archives to create, manage, and share archival information more efficiently.

For the sake of interoperability, the selection criteria for a commercial archival management system must include the ability to import and export EAD files, ideally both one at a time and as a batch process. Commercial packages provided by Adlib, CALM, CuadraStar, and Eloquent Systems all provide batch and individual import and export of EAD finding aids. If your institution requires a hosted solution, many vendors offer such an option.

Use an XML publishing platform

XML publishing platforms enable documents to be searched, browsed and displayed. Implementing them requires fairly sophisticated systems administration and programming knowledge. Some XML publishing platforms are listed in Appendix II.

EAD can be complicated (but there's hope...)

EAD was designed to be flexible in order to accommodate a broad range of archival practice. In offering flexibility, the standard has succeeded almost too well.

Freedom of choice in implementation means, for example, that three people could encode the extent of a collection in three different ways. This flexibility in implementation can cause difficulties for aggregators who harvest EAD data from multiple institutions for indexing and searching. It also hinders tool development since tool builders must either allow for multiple encoding options or choose one "right way," when there are multiple correct ways to encode the same thing. EAD's inherent complexity makes it difficult for institutions to make decisions regarding implementation. Those who are choosing tools must evaluate the choices made by tool builders to ensure that outputs meet their own best practice guidelines.

So what to do? Make a decision. Document the decision. Apply it consistently. Until the flexibility inherent in EAD is in reigned in, institutions can maximize the consistency of their data by:

1. Selecting a template in use at one or more institutions, or creating a template that adheres to a "best practice" document in use at one or more than one institution. Once you've established a template, deviate from it as little as possible.

2. Clearly document how dates, extent, etc., should be encoded. Follow your own documentation rigorously.

3. Refrain from excessively complex coding (for example, nesting duplicate `scopecontent` elements within each other).

4. Refrain from adding unnecessarily elements such as `list` elements within a control access simply to achieve a desired appearance in the output. EAD should be only be used to encode the structure and content of a document; appearance should be controlled by the stylesheet.

Remember, too, that the entire EAD tagset need not be used. As mentioned above, limiting yourself to collection-level descriptions and the DACS minimum-level description elements can simplify EAD immensely.

Getting through it

Despite a more than a decade of practice, archivists still encounter significant barriers in EAD implementation. We hope this paper gives you options to get over hurdles, under obstacles, around complexity, and through difficulty.

We recognize that EAD can be challenging. Examples of EAD's complexity can be found easily by looking through the EAD Tag Library. Many elements, including `accessrestrict`, `controlaccess`, `bioghist`, and `note`, may be repeated within an element with the same name to an arbitrary depth; for example, EAD allows one to encode nested `controlaccess` elements with no restrictions on how deep that nesting goes. In addition, EAD has seventeen linking elements; of those seventeen, twelve of those elements allow the `href` attribute, which allows linking to resources external to a given EAD file. Elements that allow "mixed content" (those that can contain both text and other elements in arbitrary order) can present problems when importing EAD to a database or porting to another data scheme. Some elements that can be full of mixed content and contain information that would be lost in migration to a database (or would require additional tagging after export) are `p`, `listitem`, `bibref`, and `head`.

EAD will be under active review in the near future. We recommend that the Technical Subcommittee for EAD (the soon-to-be charged successor to the EAD Working Group) and the archival community as a whole consider ways in which EAD can be simplified.

As reflected in the large number of EAD tools listed in this paper and its appendices, there are many choices for would-be EAD implementers. While diversity and choice is a good thing, the range and number of available tool choices provide additional complexity. By highlighting tools that are already available, we encourage institutions to utilize work that has been done elsewhere and not to invest what might be unnecessary development effort.

Acknowledgements

Appendix I. Consortia and EAD Aggregators

Almost all can provide a means of publishing finding aids, or may serve as an additional distribution channel for collection descriptions. Many also have tools to aid in EAD creation, provide instruction opportunities, and have developed best practice guidelines.

United States

- **Archival Resources in Wisconsin:** http://digital.library.wisc.edu/1711.dl/wiarchives

- **Archives Florida:** http://palmm2.fcla.edu/afl/

- **Arizona Archives Online:** http://azarchivesonline.org

- **Black Metropolis Resources Consortium:** http://www.blackmetropolisresearch.org/ [*forthcoming*]

- **Historic Pittsburgh:** http://digital.library.pitt.edu/pittsburgh/

- **Kentuckiana Digital Library:** http://kdl.kyvl.org/

- **Mississippi Digital Library:** http://msdiglib.net/

- **Mountain West Digital Library:** http://mwdl.org/index.php/search/results?format=ead

- **Northwest Digital Archives:** http://nwda.wsulibs.wsu.edu

- **OhioLINK:** http://ead.ohiolink.edu

- **Online Archive of California (OAC):** http://www.oac.cdlib.org/

- **Pennsylvania Digital Library:** http://padl.pitt.edu/

- **Rhode Island Archival and Manuscript Collections Online (RIAMCO):** www.riamco.org [*forthcoming*]

- **Rocky Mountain Online Archive:** http://rmoa.unm.edu

- **Texas Archival Resources Online (TARO):** http://www.lib.utexas.edu/taro

- **Virginia Heritage Project:** http://www2.lib.virginia.edu/small/vhp/

UK and Continental Europe

- **A2A** (Access to Archives, United Kingdom): http://www.nationalarchives.gov.uk/a2a/

- **Archives Hub** (United Kingdom): http://www.archiveshub.ac.uk/

- **Archives Portal Europe:** http://www.apenet.eu/ [*forthcoming*]

- **MALVINE** (Europe): http://www.malvine.org/

- **National Archival Database of Sweden:** http://nad.ra.se/static/back_eng.html

Subject based

- **Navigational Aids for the History of Science in Europe**
  (NAHSTE): http://www.nahste.ac.uk/

- **Guide to Australian Literary Manuscripts:** http://findaid.library.uwa.edu.au/

- **Irish Literature Collections Portal:** http://irishliterature.library.emory.edu/

- **Physics History Finding Aids Web site**
  (PHFAWS): http://www.aip.org/history/nbl/findingaids.html

Other

- **ArchiveGrid:** http://www.archivegrid.org
  OCLC's Archive Grid combines finding aids with MARC records to create one-stop-shopping
  for users. Heavy representation from US institutions, also representation from outside the US.
  Contribution is free and open to any institution.

Appendix II. Tools

Templates

Examples of templates include:

- **Northwestern University:** http://staffweb.library.northwestern.edu/dl/ead/template.xml

- **Syracuse University:** http://library.syr.edu/digital/guides/ead/aaa_template.xml

- **Indiana University Bloomington:** http://www.dlib.indiana.edu/services/metadata/activities/eadDocumentation.shtml

Web-based forms

Following are a few examples of Web-based forms. Some produce only collection-level EAD (that is, they do not include the inventory in the `<dsc>` section), while others produce complete inventories. They differ in ease of use, complexity, and quality of product. The first two are probably the most complete.

- **California Digital Library:** http://www.cdlib.org/services/dsc/tools/ead_webtemplates.html
  The California Digital Library has created numerous online templates that create EAD from typed or copy-pasted data. One or more of these may be adaptable for your institution's needs.

- **ArchivesHubUK:** http://www.archiveshub.ac.uk/eadform2002.html
  Web-based form, renders complete EAD document. Enables editing of uploaded files, creation of new ones, saving of draft file between editing sessions, preview feature. Links to digital surrogates are easy to add, common markup tags (e.g. paragraphs, lists, titles, links) can be added via a right-click menu, and a special characters keyboard is provided. Components can easily be added to create sub-fonds descriptions, and a tree structure will show exactly what the hierarchy looks like.

- Northwestern
  University: http://staffweb.library.northwestern.edu/dl/ead/eadchef/template.cgi/ead/nul

- Notre Dame: http://classic.archives.nd.edu/ead/ead.htm
  includes both collection level and inventory forms

- Berkeley: http://sunsite3.berkeley.edu/ead/tools/template/
  cgi Web application; appears to be collection-level only. Last update to page is 2005.

- Western Kentucky: http://pax.uky.edu/template-v1-cgi/template.pl/KNVUA_generic.tmp

- Austin College Xforms tool: http://www.archivists.org/saagroups/ead/tools.html
  Developed by Justin Banks at Austin College. Requires server that supports Xforms.
  Additional information from SAA 2007 available
  here: http://matienzo.org/saa2007descriptionexpo.

Standalone forms

- The University of Utah has a java-based EAD authoring tool called xEAD, currently publicly
  available at https://lsta.lib.byu.edu/lstawiki/index.php/XEAD_Project. The application opens
  an EAD file in its buffer, allowing users to manipulate the data and then resave.

- The German Bundesarchiv, with assistance from the Mellon Foundation, has developed a
  tool called Midosa Editor for XML or MEX. It is available in English and German and for both
  OSX and Windows. MEX provides an authoring environment for creating a variety of levels of
  EAD records and includes built-in publication to HTML capability. Background information is
  available at http://www.bundesarchiv.de/daofind/en. Downloads and a quick-start guide are
  available from the MEX SourceForge wiki at http://mextoolset.wiki.sourceforge.net/.

Style sheets for authoring finding aids

- Yale University has developed the Yale Finding Aids Creation Tool (FACT), which is a
  customized version of XMetaL Author. The tool includes a set of style sheets, macros, and
  other customizations. More information is available at http://yalefact.pbworks.com.

- Northwestern Digital Archives: http://nwda.wsulibs.wsu.edu/tools/ead.css

- Syracuse University: http://library.syr.edu/digital/guides/ead/ead.css

XSLT stylesheets for displaying finding aids

- **EAD 2002 Cookbook:**
  http://www.archivists.org/saagroups/ead/ead2002cookbookhelp.html

- **EAD Help pages**' user contributed style sheets:
  http://www.archivists.org/saagroups/ead/stylesheets.html

- **UC Berkeley EAD Tools:** http://sunsite3.berkeley.edu/ead/tools/

- **NC Echo EAD Tools:** http://www.ncecho.org/dig/ncead.shtml#tools

- **University of Minnesota:** https://wiki.lib.umn.edu/Staff/FindingAidsInEAD

Commercial XML tools for EAD encoding

- **oXygen:** http://www.oxygenxml.com/
  Commercial XML authoring and editing software. Data entry, copy-paste; can handle DTD or schema

- **XMetaL:** http://na.justsystems.com/content-xmetal
  Commercial XML authoring and editing software. Data entry, copy-paste; can handle DTD or schema.

- **XMLSpy:** http://www.altova.com
  Commercial XML editing, authoring, development environment.

- **NoteTab Pro:** http://www.notetab.com/
  General purpose text editor which can be easily customized to handle EAD. Clip libraries and other add-ons are available here http://www.archivists.org/saagroups/ead/tools.html

Content Management Systems for Archives

- **Archon:** http://www.archon.org
  Open source full archival management system, developed by University of Illinois with funding from Mellon Foundation. Capable of ingesting MARC or CSV format. Can export EAD and MARC.

- **Archivists' Toolkit:** http://www.archiviststoolkit.org
  Open source full archival management system, developed by UCSD, NYU, and Five Colleges

Inc. and funded by Mellon Foundation. Capable of ingesting tab-delimited fields and MARC as well as preexisting EAD. Can export EAD, Marc, Dublin Core, MODS, METS.

- **MSAccess, FileMakerPro, etc.**
  Commercial but reasonably priced general database development tool. A database developed in-house with the appropriate fields, in conjunction with XSL, java, or other scripting languages, can generate EAD.

Archival management systems that support publishing

- **Adlib** (commercial): http://www.adlibsoft.com/

- **Archon** (open source): http://archon.org/

- **Calm** (commercial): http://www.crxnet.com/page.asp?id=57

- **Cuadra STAR/Archives** (commercial): http://www.cuadra.com/products/archives.html

- **Eloquent** (commercial): http://www.eloquent-systems.com/products/archives.shtml

- **ICA-AtoM** (open source): http://ica-atom.org/ *[currently available in Beta release]*

- **Minisis M2A** (commercial): http://www.minisisinc.com/index.php?page=m2a

Various other commercial archival management systems can import and export EAD, including:

- **Minisis**: http://www.MINISISinc.com

- **Eloquent Archives:** http://www.eloquent-systems.com/products/archives.shtml

- **AdLib:** http://www.adlibsoft.com

- **CALM:** http://www.ds.co.uk

For detailed information on these and other commercial tools, refer to Lisa Spiro's 2009 report for CLIR, Archival Management Software (http://clir.org/pubs/reports/spiro2009.html), or to the associated wiki at http://archivalsoftware.pbwiki.com/.

XML publishing platforms[45]

- **XTF:** http://xtf.wiki.sourceforge.net/
  "A flexible indexing and query tool that supports searching across collections of heterogeneous data and presents results in a highly configurable manner." XTF supports powerful searching, faceted browsing, and viewing search terms in context.
  (Open source; used by California Digital Library and numerous others)

- **Mark Logic:** http://www.marklogic.com/
  The University of Chicago is developing an XML publishing infrastructure built on MarkLogic, a native XML database. The front end can be built on any platform and provides flexible display options. The University of Chicago's code will be available to anyone. Archives that want to use the software will need MarkLogic, but there is a free (limited) version that will suffice for small institutions.
  (Commercial; used by University of Chicago)

- **PLEADE:** http://www.pleade.org/en/
  "open source search engine and browser for archival finding aids encoded in XML/EAD. Based on the SDX platform, it is a very flexible Web application."
  (Open source; used by Denver Public Library)

- **Cocoon:** http://cocoon.apache.org/
  Cocoon is an open source XML publishing framework that applies XSLT stylesheets to the EAD finding aid to display HTML. Used in tandem with indexing technologies such as Lucene or eXist.
  (Open source; used by Ohio State and the Five College Finding Aids Access Project)

- **DLXS:** http://www.dlxs.org
  XML-aware search engine (XPAT or XPAT Lite) with DLXS middleware which includes a "class" for finding aids (currently in prototype).
  (Open source component, with commercial options available; used by University of Michigan and University of Minnesota Libraries[46])

- **Cheshire3:** http://www.cheshire3.org/
  "Fast XML search engine." Standards compliant, with support for Open Access Initiative (OAI) protocols and Z39.50. Modular and configurable."
  (Open source, used by University of Liverpool Special Collections and Archives and ArchivesHub)

Specialized migration or conversion tools

- **MARC to EAD—MarcEdit:** http://oregonstate.edu/~reeset/marcedit/html/index.php
  If you have MARC records for your manuscript collections, you can quickly and easily generate skeleton (i.e., collection-level) EAD records from it using MarcEdit. MarcEdit uses xsl style sheets which can easily be modified/customized. Developed by Terry Reese at Oregon State University; free.

- **Excel to EAD using Mailmerge**
  Excel is an immensely useful tool for generating the code for lengthy inventories of minimal depth. Text can be entered into a spreadsheet, then columns can be added before and after the text and populated with the correct EAD elements. For collections with large inventories, either copy-pasting (from Word, RTF, or txt) or entering afresh in Excel may be a workable solution. Excel's MailMerge feature can also be employed to automatically generate coded data from an Excel spreadsheet; see video
  here http://archives.state.ut.us/containerlist/containerlist.html. This could be used in conjunction with MarcEdit which generates the collection-level part of EAD to produce a full EAD inventory. Indiana University has posted detailed instructions and an Excel template for encoding lengthy inventories
  at http://www.dlib.indiana.edu/services/metadata/activities/eadDocumentation.shtml (see the section "Using Excel to assist with encoding").

- **Text to EAD—EAD Conversion:** http://agileimage.com/html/ead/
  Reads a text version of inventory and generates an EAD-encoded version. Last update to Web site was in 2004.

- **Text to EAD—MSWord**
  Starting with an electronic file of an inventory, a surprising amount of tagging can be done simply using MSWord's search and replace feature with tabs and regular expressions, including locating and tagging `unitdate`, `extent`, long inventories, etc. Good overviews of regular expressions in word are available here:

- http://office.microsoft.com/en-us/help/HA010873051033.aspx

- http://office.microsoft.com/en-us/help/HA010873041033.aspx

- http://word.mvps.org/FAQs/General/UsingWildcards.htm

- Date normalization—tri-XMLdate-normalizer.pl: http://www.archivists.org/saagroups/ead/tools.html
  Developed by Jason Casden at The Ohio State University; free. For large files, inserting the `normal` attribute for `unitdate` elements can be extremely time-consuming. This Perl script automatically recognizes numerous date formats and inserts the `normal` attribute. Offers options to overwrite existing values or leave them.

Papers, production guides, case studies, etc.

- **Northwest Digital Archives Standards Working Group Review of Web Templates for EAD Creation** (October 2008): http://www.orbiscascade.org/index/cms-filesystem-action?file=nwda/files/bowmanreview_200810.pdf – excellent and thorough review of available Web-based templates that generate EAD.

- **OhioLINK EAD Starter Package** (April 2008): http://platinum.ohiolink.edu/dms/ead/contentguide/EAD_starter_packet_v4.pdf – Draft of paper looking at steps in an EAD conversion project; very rough but good info.

- **University of Indiana EAD guide:** http://www.dlib.indiana.edu/services/metadata/activities/EADManual.pdf – includes detailed instructions for using oXygen.

- **Utah State Archives EAD Project:** http://www.archives.state.ut.us/research/inventories/ead.html – includes detailed discussion of how they converted their legacy finding aids using a combination of tools (HTML, Excel, WordPerfect, etc).

- **NYU Archives EAD Production Guide:** http://www.nyu.edu/library/bobst/research/arch/eadProduction.htm – detailed procedures used by NYU, includes detailed instructions for NoteTab Pro.

- The **EAD Help Pages:** http://archivists.org/saagroups/ead/sitesann.htm – more than 80 institutions currently implementing EAD, including brief descriptions of each institution's approach and a point-of-contact. Implementations run the gamut from extremely simple (EAD put online with a style sheet) to extremely sophisticated (databases that provide server-side transformations and advanced search capabilities).

Appendix III. EAD Migration, Creation and Publication Paths

Review best practice, local practice

Write up tagging specifications

Source file format?

- Database → **Work with it to export tagged data**
- Hard copy → **OCR or re-key**
- Text/word processing → **E-text**

Got MARC?

- Yes → **Export collection-level EAD from MARC**
- No → **Level of tool desired**

Level of tool desired

- Full collection management → **Select collections management tool** → **Import EAD created from database or MARC (above)**
- Just EAD → **Select XML authoring tool(s)**

May include one or more of the following:
- Web-based template
- XML authoring application
- Scripts

Or

Edit, update, or create EAD files to specification

Create or update MARC records
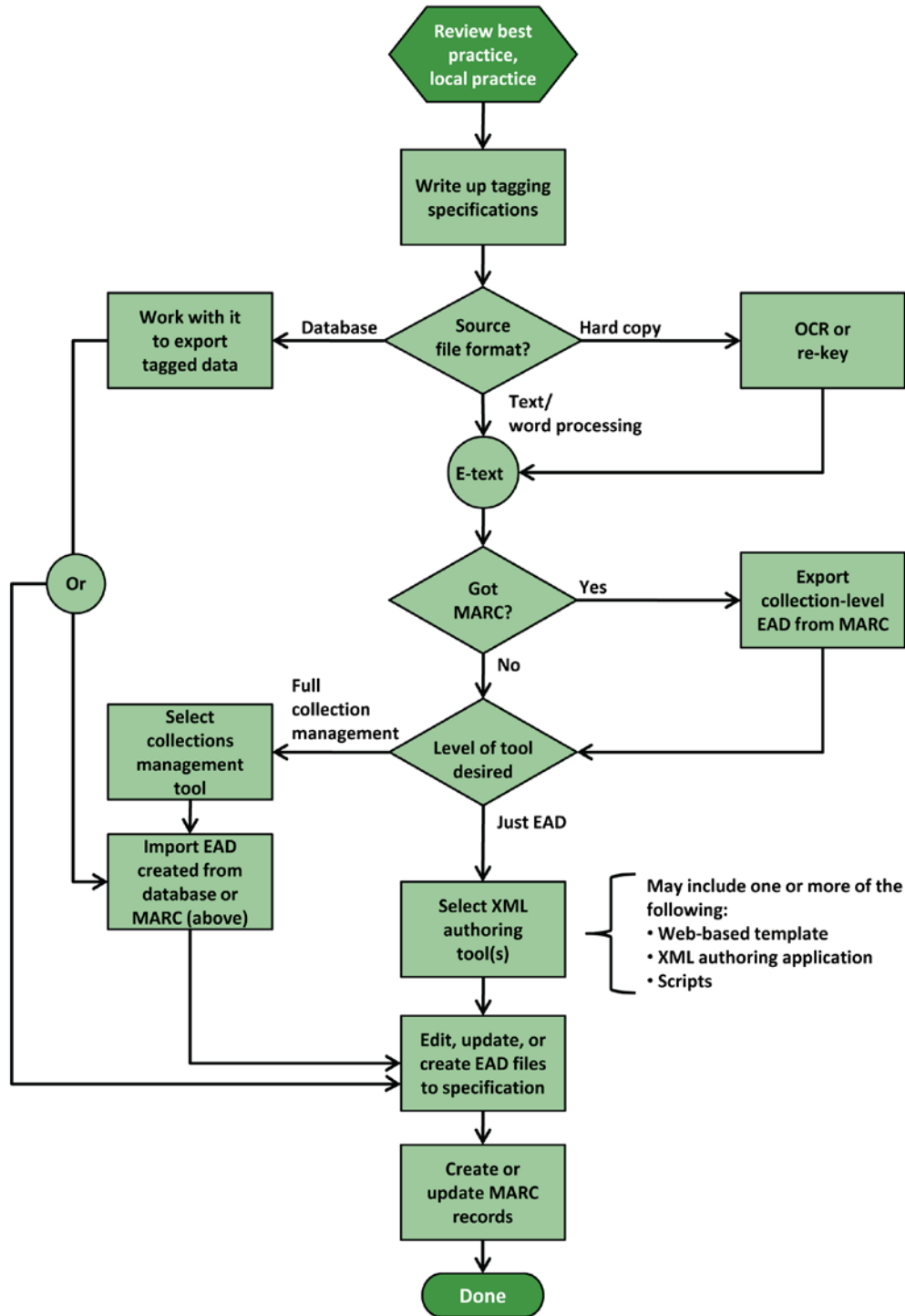
Done

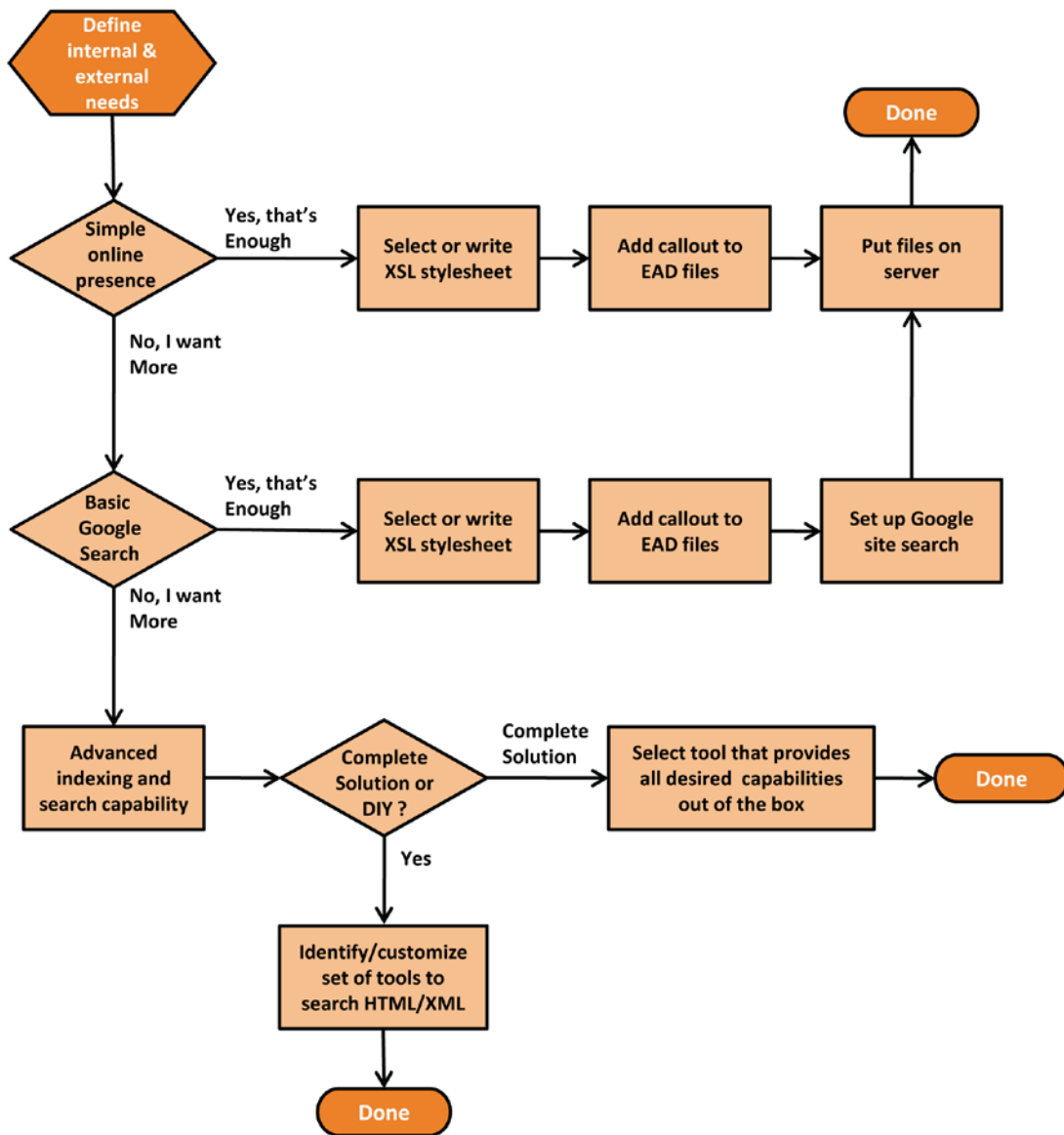## Figure 1. EAD Migration and/or Creation

Figure 2. EAD Publishing Paths

Simplest path (conversion from MARC records)

5. Export collection-level EAD records from MARC using MarcEdit.

6. Validate and correct errors using W3C's online validation tool (http://validator.w3.org).

7. Select or create an XSL style sheet.

8. Put files on server.

Simplest path (starting without MARC)

9. Use one of the many existing Web-based templates to generate collection-level EAD files.

10. Validate and correct errors using W3C's online validation tool (http://validator.w3.org).

11. Find or create an XSL style sheet.

12. Put files on server.

The above solutions require minimal time, expertise, and money, yet yield online collection-level descriptions that will be "crawled" by Google and other search engines making them discoverable via the open Web, and valid EAD files which can be contributed to consortia (a list of consortia can be found in Appendix I) or aggregators like OCLC's ArchiveGrid. Files can easily be "upgraded" at some later date without reworking.

More sophisticated path

13. Choose one of the full collection-management packages such as Archivists' Toolkit or Archon.

14. Perform data entry and/or import to level desired.

15. Install and configure XTF installation for Web availability, browsing, searching, etc.

16. Export EAD to be indexed/searched by XTF system.

This yields a fully-functioning database and sophisticated search capabilities but requires substantial technical knowledge to install and configure the XTF installation.

Notes

All Web links were verified 25 February 2010.

[1] EAD Roundtable, *EAD (Encoded Archival Description) Help pages*, sponsored by the Society of American Archivists, http://www.archivists.org/saagroups/ead/. In particular, see the section on "What is EAD," at http://www.archivists.org/saagroups/ead/aboutEAD.html.

[2] Archivists Toolkit User Group, "2008 AT User Group Survey Results," http://archiviststoolkit.org/sites/default/files/AT%20User%20Group%20SurveyResultsFD.pdf. Additional breakdown of survey results supplied by Brad Westbrook to the working group via e-mail on 18 December 2008.

[3] Tatem, Jill, "EAD: Obstacles to Implementation, Opportunities for Understanding," *Archival Issues* 23,2, (1998): 155-169; Roth, James M., "Serving Up EAD: An Exploratory Study on the Deployment and Utilization of Encoded Archival Description Finding Aids," *The American Archivist*, 64,2 (2001): 214-237; Dow, Elizabeth H, "EAD and the Small Repository," *The American Archivist*, 60,4 (Fall 1997): 446-455.

[4] Wisser, Katherine M., *EAD Tools Survey*, Society of American Archivists, EAD Roundtable, (August 2005), http://www.archivists.org/saagroups/ead/EADToolsSurvey.pdf; Yaco, Sonia, "It's Complicated: Barriers to EAD Implementation," *The American Archivist*, 71,2 (Fall/Winter 2008): 456-475.

[5] A recent report from CLIR documents the current state of archival management software, much of which provide support for EAD: Lisa Spiro, *Archival Management Software, A Report for the Council on Library and Information Resources*, CLIR Reports (January 2009), http://www.clir.org/pubs/reports/spiro2009.html.

[6] Children's Television Workshop, *Sesame Street*, Episode 0299 (December 9, 1971).

[7] Yaco, "It's Complicated," 468: "another key barrier is 'Lack of institutional support'."; Wisser, *EAD Tools Survey,* 19: "Some folks in the organization... are unconvinced of EAD's staying power and

thus stingy when it comes to allocating resources and time for relevant training. Is there an EAD eye-opener kit?"

8 International Council on Archives. ISAD(G): General International Standard Archival Description, Second Edition, http://www.ica.org/en/node/30000; Society of American Archivists, Describing Archives: A Content Standard (DACS), http://www.archivists.org/governance/standards/dacs.asp.

9 Society of American Archivists. "Appendix A: EAD Crosswalks," Encoded Archival Description Tag Library, Version 2002, The Library of Congress, http://www.loc.gov/ead/tglib/appendix_a.html.

10 Spiro, Archival Management Software.

11 U.S. Library of Congress Network Development and MARC Standards Office, EAD (Encoded Archival Description) Electronic List, http://www.loc.gov/ead/eadlist.html; Society of American Archivists, Web site, www.archivists.org; Rare Book School, Web site, http://www.rarebookschool.org/.

12 Szary, Richard V., "Encoded Finding Aids as a Transformative Technology in Archival Reference Service," in Encoded Archival Description on the Internet, ed. Daniel V. Pitti and Wendy M. Duff, 187-197. Binghamton, NY: Haworth Information Press (2001).

13 ArchivesZ, http://archivesz.com/; Syracuse University Library, "Elastic Lists: EAD demo", http://library.syr.edu/digital/guides/ElasticLists-EAD/binv3/index.html; NNDB Mapper, http://mapper.nndb.com/.

14 National Endowment for the Humanities, Division of Preservation and Access, "Frequently Asked Questions: Humanities Collections and Resources," http://www.neh.gov/grants/guidelines/collections&resfaqs.html.

15 National Information Standards Organization, A Framework of Guidance for Building Good Digital Collections, 3rd Edition, with support from the Institute for Museum and Library Services (IMLS), http://framework.niso.org/node/38.

16 National Historical Publications and Records Commission (NHPRC), "Basic Projects Grant Announcement," http://www.archives.gov/nhprc/announcement/basic.html, and "Detailed Processing Projects Grant Announcement," http://www.archives.gov/nhprc/announcement/detailed.html.

17 EAD Roundtable, "90 Implementors Currently Listed," EAD Help Pages http://www.archivists.org/saagroups/ead/implementors.html.

18 Yaco, "It's Complicated," 459: "Concern about the quality or completeness of finding aids often causes archivists to rewrite legacy finding aids before EAD encoding."

www.oclc.org/research/publications/library/2010/2010-04.pdf
Combs, et al., for OCLC Research
February 2010
Page 42

[19] Voltaire, *La Bégueule: Conte moral*, (1772).

[20] Mark A. Greene and Dennis Meissner, *More Product, Less Process: Pragmatically Revamping Traditional Processing Approaches to Deal with Late 20th-Century Collections,* http://ahc.uwyo.edu/documents/faculty/greene/papers/Greene-Meissner.pdf, page 2. (Also published as "More Product, Less Process: Revamping Traditional Archival Processing," *American Archivist* 68,2 (2005): 208-63.)

[21] Society of American Archivists, "Code of Ethics for Archivists," http://www.archivists.org/governance/handbook/app_ethics.asp.

[22] Society of American Archivists, "ALA-SAA Joint Statement on Access: Guidelines for Access to Original Research Materials," http://www.archivists.org/statements/alasaa.asp.

[23] Yaco, "It's Complicated," 466: "lack of staff...[along with] lack of time."

[24] "MarcEdit—Your Complete Free MARC Editing Utility," (Web site hosted at Oregon State University) http://people.oregonstate.edu/~reeset/marcedit/html/index.php.

[25] Yaco, "It's Complicated," 460: "the complexity of EAD itself [is] a deterrent to implementation."

[26] Yale University Library FACC: Finding Aids Coordinating Committee, "Yale University EAD Encoding Best Practice Guidelines," http://www.library.yale.edu/facc/bpgs.html.

[27] Yaco, "It's Complicated," 466: "Lack of a plan".

[28] Society of American Archivists, Encoded Archival Description Working Group, *Encoded Archival Description Application Guidelines for Version 1.0*, (Encoded Archival Description (EAD), Document Type Definition (DTD), Version 1.0, Technical Document No. 3), (1999), http://www.loc.gov/ead/ag/aghome.html.

[29] Spiro, *Archival Management Software,* 6: "Archival material is so specific that you don't want to get locked into anything...Ideally, I would want something that would also preserve that information in a format that is able to migrate if needed."

[30] RLG EAD Advisory Group, *RLG EAD Best Practice Guidelines for Encoded Archival Description*, (August 2002), http://www.oclc.org/programs/ourwork/past/ead/bpg.pdf.

[31] Michele Combs, *EAD Tagging Specs (Revised 10-3-08),* Syracuse University Library, http://library.syr.edu/digital/guides/ead/tagging_specs.doc.

[32] World Wide Web Consortium (W3C), *Markup Validation Service*, http://validator.w3.org.

www.oclc.org/research/publications/library/2010/2010-04.pdf
Combs, et al., for OCLC Research
February 2010
Page 43

[33] Spiro, *Archival Management Software,* 12: "Some institutions, however, lack the technical staff to implement open source software. Others may oppose it because of they fear security risks or high maintenance costs."

[34] Archon and the Archivists' Toolkit are both open source applications. At this writing, the two groups are looking into the possibility of combining the best features and functionality of each application into a single application.

[35] Koha, http://www.koha.org/; Evergreen, http://open-ils.org/.

[36] Fedora, http://fedora-commons.org/; DSpace, http://dspace.org/; EPrints, http://eprints.org/; Greenstone, http://www.greenstone.org/.

[37] Yaco, "It's Complicated," 461: "Even archives with the resources to create custom, sophisticated encoding computer programs struggle with publishing issues."

[38] Refsnes Data, w3schools.com, "Displaying XML with XSLT," *XML Basic,* http://www.w3schools.com/Xml/xml_xsl.asp.

[39] Refsnes Data, w3schools.com, "Displaying XML with CSS," *XML Basic,* http://www.w3schools.com/Xml/xml_display.asp.

[40] Bob DuCharme, "Using XSLT to Deliver XML on Browsers," *bobdc.blog,* http://www.snee.com/bobdc.blog/2008/09/using-xslt-to-deliver-xml-on-b.html.

[41] Syracuse University Library, Special Collections Research Center, http://library.syr.edu/information/spcollections/.

[42] Swish-e, http://swish-e.org/.

[43] Google Inc., Google Site Search, http://www.google.com/sitesearch/.

[44] For more information on archival management systems, see the Archival Software wiki, http://archivalsoftware.pbwiki.com/FrontPage.

[45] This section adapted from Spiro, *Archival Management Software.*

[46] University of Minnesota Libraries, "Finding Aids in EAD," *Libraries Staff Wiki,* https://wiki.lib.umn.edu/Staff/FindingAidsInEAD.