FINDING A NEW WAY:
USING AUTOMATED BUSINESS RULES TO PROCESS ELECTRONIC RECORDS

Richard Pearce-Moses
Society of American Archivists
Austin, Texas 2009

The Arizona State Library and Archives is the lead partner on the PeDALS Project. PeDALS stands for Persistent Digital Archives and Library System.

The PeDALS project partners include the Florida State Library and Archives, the South Carolina Department of Archives and History and the South Carolina State Library, the New York State Library and the New York State Archives, and the Wisconsin State Archives. And, we're looking for two more partners. To give credit where credit is due, the project is funded principally through the Library of Congress National Digital Information and Infrastructure Preservation Program and the Institute for Museum and Library Services Library Services and Technology Act. I always like to acknowledge the Washington State Archives, which helped me conceptualize this project.

The project's goals include

- To develop a rationale for curating large collections of electronic records and digital publications
- To test the use of middleware to automate the processes underlying that curatorial rationale
- To test the use of LOCKSS as a distributed preservation storage system
- To build a collaborative community of shared practice
- To work as inexpensively as possible

PeDALS is a research project. We have great faith that it will produce a working system that we can sustain beyond the project. We're not in this just to get new insights into electronic records systems. To the contrary, we're looking to understand what it means to apply the theoretical knowledge that others have already discovered.

Based on our experience, we have already proposed a new research agenda to the Library of Congress.

- Cloning. We could all use more staff. But finding the right staff is really hard. So rather than investing in time doing a search and the risks of finding out too late that the candidate isn't right for the job, we want to be able to take a good employee and make multiple copies.
- Time travel. Things seem to always take longer than you expected. With time travel, you can take all the time you need to meet a deadline at a relaxed pace, and then you just rest the clock.
- Vulcan mind meld. One of the most difficult challenges we faced is helping people understand the vision for the system. Who needs fallible words when you have telepathy!

We sincerely hope that someone will get positive results soon! Seriously, we have already had a number of insights that I hope you will find useful.

A bit more on PeDALS project. No single approach will work for every type of record, and the PeDALS system has its limitations.

PeDALS is an implementation of the Open Archival Information System reference model. At a very abstract level, the system does three things: when records are submitted, it

1. Captures the metadata necessary for administration, discovery, and preservation
2. Creates an archival information package and stores it in a LOCKSS distributed preservation storage system
3. Creates a dissemination package and publishes it on a public website.

Because PeDALS requires a certain amount of programming to automate processing, the cost of programming must be justified by the number of records the system will process. As such, PeDALS is designed for acquisitions containing many records that are sufficiently similar to be processed in a routine manner. One collection we're working with consists of more than 20,000 marriage certificates; to create an index manually would take hours, but should take a programmer relatively little time. The records may be received in a single acquisition, running the rules only once – say the Governor's email. Or they may be acquired in small batches over a longer period of time, running the rules many times.

PeDALS also assumes the records come with metadata. The ideal model is a group of records that the provenance keeps in an automated recordkeeping system. The provenance needs to index the records at some level to support their own business needs. The records and the index are delivered to the system, and the rules normalize and enhance the supplied metadata so that the system has sufficient information for administration, discovery, and preservation.

PeDALS may not work if there is no supplied metadata. For example, the governor's hard drive has many files, but no metadata beyond the operating system's filename and timestamps. Because these files are typically ad hoc, they vary significantly. They may include reports, PowerPoints, speeches, memoranda, and other formats on many diverse topics. The lack of consistency makes it difficult to write business rules to describe the records adequately. However, we have visions of using natural language processing tools to generate metadata in these cases.

Today, I'd like to share with you some early lessons we've learned. I want to touch on three broad themes. A curatorial rationale, an integrated workflow, and automating the workflow.

Curatorial Rationale

Some of the lessons revolve more around *why* we do some things than *how* we do them. As we reconceptualize how we work in a virtual system, we find that we need to make some strategic shifts. We must do more than do things differently; we must do different things.

I think the biggest shift is that we will no longer process records directly. Rather, we will work with rules that guide processes applied to records. To use Helen Tibbo's phrase, no more document fondlers. In the virtual realm, our many actions and decisions are based on queries against the metadata in a catalog. We do not sort physical objects; we instruct the system to return the information in a certain order. We do not select materials for preservation by inspection, but through a query into the metadata to find objects with specific characteristics, such as a file format that has become obsolete.

The project has developed a dictionary of core metadata that includes elements for administration, discovery, and preservation. The elements are defined in terms of the nature of the information, but also how that information is used in the system.

For example, traditional bibliographic description dictates "transcribe the title proper exactly as to wording, order, and spelling, but not necessarily as to punctuation or capitalization."[1] Archivists are familiar with creating titles when there is none, but how do you do that when using a machine – rather than a human – to supply the title? For marriage certificates, we could use that phrase as the title. However, the use of an obvious title raises problems for retrieval. Consider the results of a search if you're presented with several hundred records described identically as "marriage certificates." We need a title that is more than a name; we need a title that aids in the selection of the material. And that's the *why* of the process. The solution is to write a rule that supplies a title that includes the bride's and groom's names as well as the form "marriage certificate" so that search results are meaningful.

[1] AACR2R, 1.1B (1988).

Integrated Workflow

We are trying to articulate a curatorial rationale that looks at the complete range curatorial functions from submission to archival storage and dissemination. A great deal of research in electronic records has focused on a fairly narrow curatorial function, such as the impact of format shift on meaning, methods for rendering obsolete information formats, and the like. This research is invaluable, but it's often hard to know how to apply it in context.

We have developed a "swimlane" diagram that groups activities in the workflow. As we build the system, we often discover steps that are missing, that need to be made more granular, or that need to be refined.

Creation: The provenance must prepare the submission information package. That includes identifying records for transfer, exporting the records and accompanying metadata in an acceptable format, and generating a shipping manifest. In our experience, this work needs to be as simple as possible. The record creators are busy and have limited resources. If the archives asks the provenance to do too much, the work may not be done at all. The field archivist needs to know enough about the system and its technical requirements to be able to negotiate a good compromise between what the provenance can provide and the minimum requires to be able to process the records. Our basic philosophy is, We take what we can get.

Description: The archives staff begins the process of describing the collection, capturing the information necessary for administration, discovery, and preservation. The workflow begins with fairly traditional activities: creating an entry for the provenance, including an administrative history and variant names; describing the series, assigning a title, writing a scope note, and assigning access points; finally, each specific acquisition is described.

At this point, description shifts from the records to process. The archivist maps the available metadata to the core schema. The map is a set of rules that describes how received metadata is to be transformed, normalized, and enhanced so that it can be stored in a standard schema. After the archivist maps the metadata, the programmer rewrites those rules in software.

Description also anticipates storage and dissemination by writing the rules to encapsulate the records in standard archival and dissemination information packages schemas.

Submission: With the prep work done, it's time to begin working with the records. The first step is the tranferring the records from the provenance to the system. Again, it's important to keep this simple for the provenance, so we'll effectively take the records however we can get them. We can work with anything from secure ftp to sneaker net.

The Point of Ingest is a server that is in front of a firewall. We want to isolate the records and do a preliminary virus scan before they're brought into the system.

We also use that system to do some initial quality control. In the best of all worlds, we'd get records in a form we expected, in the manner we negotiated with the provenance. In reality, we've discovered a need for "data wrangling." I use this phrase to distinguish manual processes necessary to ensure that the records are in a form that won't trip up subsequent automated systems.

The first few times you ingest records, you're going to discover assumptions the provenance and you made. You may be surprised that the metadata is in all capital letters. You've learned that some of your tools choke with very large record sets, and you need to split the records into smaller batches. You discover that the metadata for some records in the files is malformed.

Data wrangling should use of a variety of tools to test in different environments. We carefully selected marriage certificates because they did not contain any personal information that would present a security risk. All was right with the world as I tested things on my Vista machine. But when I tried the same exercises on my XP machine at home, I discovered that the TIFF files they delivered in fact contained more than we bargained for. The VIsta TIFF viewer can't handle multi-page TIFFs. As a result, it wasn't until I use my XP machine, which comes with a different viewer, that I discovered I was working with multipage TIFFs.

Especially the first time you bring a set of records, it is probably a good idea to spend some time characterizing the data you have to make sure it's what you planned for You thought the license

numbers would always be a long integer, but when you check the data you find out they're alphanumerics. Similarly, you may spot problems with the data. I found really odd data that looked like partial words and names in the license number field; in fact, the data in the provenance's system was corrupt for a few records. I import metadata into Access or Excel and then sort the columns to get a sense of the data. Weird things typically show up at the top or bottom of these sorts.

Data wrangling can be an interesting challenge because it's so unpredictable; it's a chance for detective work and creative problem solving.

AIP Processing: When data wrangling is complete, the rest of the process is effectively automated. The archivists initiates processing, and the middleware executes the metadata map and updates the administrative catalog. The middleware then creates the AIP, which includes the records itself, the received metadata, and the normalized metadata.

Quality Assurance: After the AIPs are created, the archivist checks to make sure the business rules ran as intended by inspecting a sample of the records. If there are problems, the business rules are tweaked, and the AIP processing rerun.

Ingest: When the AIPs are accepted, the system stores them within the LOCKSS system. We chose LOCKSS because it offers a number of features desirable within an archival storage system. First, it does automatic integrity checking and error correction through a sophisticated audit. Second, it stores multiple copies of the records on different servers, and those servers can be distributed in different locations to protect the contents against regional disasters. LOCKSS is designed to be inexpensive. Finally, the LOCKSS Alliance has a sustainable business model based on financial support from a large number of users.

Dissemination: Finally, dissemination processing parallels the AIP processing. The middleware transfers metadata from the administrative catalog to the public catalog, create a browser friendly DIP, and publishes the DIP on a website for public access, with the archivist monitoring the processes for quality assurance.

This may sounds like an enormous amount of work, and it's by no means trivial. However, by looking at the curatorial process as an integrated system, I believe we've found several ways to simplify things.

Community of Shared Practice: By working collaboratively, we believe we can work more efficiently. Rather than inventing several different systems, having similar systems offers the opportunity for resource sharing. Staff at one repository will be able to turn to others for support, which helps ensure sustainability during staff turnover. We also know there will be an opportunity for code sharing; no doubt, all the partners will have Outlook PST email files to ingest, and if we're using similar systems it will be easier to modify code.

Generic, Modular Processes: As much as possible, we are writing code for reuse. We've already demonstrated this as a proof of concept. Code written to processing Arizona marriage certificates was adapted to process South Carolina public safety commission orders in less than two hours. The process was quick because the records and metadata were similar in format (records in PDFs and metadata in XML), and that won't always be the case. But we may be able to negotiate with the provenance to get SIPs in formats we can easily process.

PeDALS Core Metadata: We decided to create a single core metadata schema to capture information common to government records, rather than separate schemas for each series. As a result, metadata mapping is a many to one relationship, rather than a many to many relationship. This cuts the amount of work in half, and it will support code reuse.

The PeDALS Core has more discovery elements than Dublin Core, plus it has elements for administration and preservation. As important, it has a content standard, describing how the elements are to be used to support a community of shared practice.

Automated Processing

We've found a number of tools that are particularly useful, and we hope that some of our work will be useful to others.

The heart of the system is the middleware, the software used to automate the processes that move the records through the system. We are using Microsoft BizTalk for our middleware. We chose BizTalk for a number of very practical reasons. First, we knew it would work; Washington State Archives used BizTalk to process archival records. Second, Microsoft is very generous to libraries, and we could get a license for very little money. Because BizTalk is a commercial platform, there is a group of programmers who are familiar with the product.

If I'm nervous about any part of the system, it's BizTalk. Programmers and consultants are expensive. It's a complicated product, and we're using only a fraction of its potential. We're just now getting to a point where we have metrics to evaluate throughput.

iRODS is another example of middleware. We did not go with iRODS for two reasons. It was just being released when we started the PeDALS product.. Also, at that time it could not work with LOCKSS, and Library of Congress specifically wanted us to test LOCKSS. At some point BizTalk will go away, and we would certainly want to consider iRODS among our options.

One of the real challenges is getting basic preservation metadata. We are using the New Zealand Metadata Extractor to capture essential information about the file characteristics, including file name and extension, size, software and version, and MIME type. We run NZME manually during the data wrangling phase. It produces an XML file containing the preservation metadata that is then consumed by BizTalk as part of the metadata normalization process.

We are very excited by the recent introduction of Bag It software, developed by the Library of Congress.[2] Bag It is a handy tool we can give a provenance to help them create a standard SIP that includes a shipping manifest that contains hash values. When the SIP is received, we can use Bag It to validate the transfer, ensuring no files were lost or became corrupt. Bag It is simple enough to use that we will run it manually as part of the data wrangling process, rather than invest in writing a BizTalk process.

We had hoped to use free, open source software as much as possible to save costs. For example, we hoped that a number of the servers could run Ubuntu Linux. Not only is Linux free, it is very lightweight and can run on older servers that can't support current versions of a more popular operating system. Recycle servers could reduce hardware costs. Unfortunately, we have been largely assimilated. BizTalk uses MS SQL Server, and the two products play much better with IIS than Apache.

We adopted the Agile/Scrum project management methodology. We set short term goals that can be delivered in a two week period. We meet every other day to check in on progress, find blocks, and help each other out. It's a nifty methodology, especially for research projects. Less time trying to write complete specs up front, especially when you know that you're not sure of what the final product should look like.

Project Status

If good things come to those who wait, this is going to be really, really good. We've run into a few snafus that have set us behind. Having project staff laid off and furloughed doesn't help, and having a consultant completely flake out slows things down, too.

I've had a lot to learn about project management and building a robust system. If I'd written the code for the administrator's catalog web interface, it would be done. But I'm a hack amateur; the staff who are building this are real programmers who worry about SQL injection viruses, front end data validation, and other tricks to make this a robust product.

---

[2] See http://www.digitalpreservation.gov/library/challenge/data-transfer.html.

We are also very much committed to iterative development. As we take a step forward, we often learn about things we could have done better in a previous step. After designing the catalog database schema, we found important changes after we installed test data.

But we have made progress.

Hardware and software: Is installed at each site.

Administrative catalog: We have largely finished this project, although we are doing some reengineering so that it will work with an improve catalog database schema.

Proof of concept: We have taken records through every step, except for dissemination. We are still in the process of integrating those steps. As I noted, we have also demonstrated the reuse of the code.

We should be further along. Each partner should be in the process of ingesting a new series every six weeks. Until some of the catalogers' tools are finished, they can't begin. We hope to have things ready to go by the end of September.

So keep watching us at http://www.pedalspreservation.org/, and say a prayer for us.