Case Study

Automating the Archives: A Case Study

CAROLE PRIETTO

Abstract: The establishment of an archival automation program requires that the archivist address issues of both a technical and a managerial nature. These issues include needs assessment, selection of hardware and software to meet identified needs, redesigning archival tasks in light of the system selected, and ongoing maintenance of the system selected. The present article discusses the issues Washington University Archives staff members faced in developing an automation program and the solutions they adopted. It concludes with a brief discussion of possible future directions for the automation program.

About the author: *Carole Prietto holds a B.A. in history from the University of California at Santa Barbara and an M.A. in history from UCLA. From 1986 to 1989 she was the assistant in the UCLA University Archives; since 1990, she has been university archivist at Washington University, St. Louis.*

MUCH DISCUSSION IN THE LITERATURE about archival automation concerns implementation of the USMARC AMC format and development of national descriptive standards.[1] This discussion is both useful and necessary, but it examines only one side of the automation issues archivists face. Without physical and intellectual control over collections, MARC records cannot be created and collections cannot be used. Discussions of the application of personal computers and commercial software packages to archival processing tasks are scarce. A search of the *American Archivist* and *Archival Issues* (formerly the *Midwestern Archivist*) going back to 1980 revealed only two such articles. In a 1990 article in the *Midwestern Archivist*,[2] Richard J. Hite and Daniel Linke outlined the use of a personal computer and WordPerfect in a team approach to processing at the Western Reserve Historical Society. In a 1991 *American Archivist* article,[3] James G. Carson outlined his repository's use of WordPerfect and Minaret.

Even more scarce are discussions of the decision-making process that results in the implementation of an automation program. My purpose here is to discuss the implementation of the automation program at the Washington University Archives. This process consisted of a number of steps over a three-year period: evaluating the existing hardware and software, selecting a new database management package,[4] installing and setting up the new software and training staff in its use, adding OCLC and NOTIS access to facilitate MARC-AMC cataloging, and, finally, adding our first MARC AMC records to a national bibliographic database. The discussion will conclude with an outline of some possible future directions for our automation program.

The Beginning of Special Collections Automation

The Washington University Special Collections Department purchased its first personal computers in 1988, two years before my arrival. At that time, the department's hardware consisted of four IBM-compatible computers and two printers. The printers were shared by way of a local area network and a network program, LANtastic. WordPerfect was chosen for word-processing needs. In addition, a database management software package was needed for archives and manuscripts processing. The program chosen was Marcon, then manufactured by AIRS, Incorporated.

My predecessor did not make automation a high priority, and the personal computer in University Archives, when it was used at all, was used for correspondence. Item-level finding aids, accession registers, and statistics were prepared, as they always had been, on a typewriter. The result was a small archives staff overburdened with clerical tasks while facing both a large backlog and a heavy reader services load. Soon after my arrival in 1990, it became

[1]For discussion of the MARC AMC format and the development of descriptive standards, see especially the papers of the Working Group on Archival Description, reprinted in the *American Archivist* 52 (Summer 1989) and 53 (Winter 1990), with extensive bibliography; and David Bearman, ed., *Toward National Information Systems for Archives and Manuscript Repositories: The National Information Systems Task Force (NISTF) Papers, 1981–1984* (Chicago: Society of American Archivists, 1987). Anne J. Gilliland, ed., "Automating Intellectual Access to Archives," *Library Trends* 36 (Winter 1988) is devoted entirely to microcomputer applications in an archival setting, as is *American Archivist* 47 (Summer 1984).

[2]"Teaming Up with Technology: Team Processing," *Midwestern Archivist* 15, no. 2 (1990): 91–98.

[3]"The American Medical Association's Historical Health Fraud and Alternative Medicine Collection: An Integrated Approach to Automated Collection Management," *American Archivist* 54 (Spring 1991): 184–91.

[4]The commercial database management packages discussed here are trademarks of their respective manufacturers. The author has no connection with any of the manufacturers whose products are discussed here.

apparent that the time had come to take a critical look at archives procedures, with an eye toward streamlining them. Automation offered a means to do this. We had the computers and the software; what we now needed was a plan to exploit our personal computer's capabilities to the fullest.

Needs assessment came first: what activities *should* be automated? The activities best suited for automation were those frequent and repetitive in nature, heavily paper-based, and involving a great deal of word processing. In examining our operations, we found the activity that best fit the above criteria was the creation of archival and manuscript finding aids. Once we decided what to automate, we had both the tools (personal computers and software) and a clear sense of what we wanted to accomplish.

Before any further progress could be realized, I had to learn to use Marcon Plus, the database package I had inherited, and then train the archives staff in its use. My strategy for learning Marcon was to find a test collection, design a database structure for that collection, enter data into Marcon, and generate a finding aid. These steps would provide the training I needed, which I could then pass on to the staff. The database file and finding aid that resulted could be used to evaluate Marcon's indexing, searching, and reporting capabilities.

The test collection was a group of audiotapes documenting Washington University's ongoing lecture program, the Assembly Series. The Assembly Series was an ideal test collection because of its size (about 1,000 tapes) and the need to increase the number of access points to the collection. The only finding aid available for the collection was a typed list of the lectures in rough chronological order; cross-indexes by speaker's name, lecture title, or sponsoring organization did not exist. Putting this information into a database structure would enable us to generate these cross-indexes easily. These indexes could easily be updated as more information was added to the database.

Unfortunately, Marcon had a number of drawbacks. The first I noticed was that the program's processing speed slowed dramatically after we entered one collection of approximately a thousand records. Over time, thousands of records would be entered into the database, and I did not want a program that would be bogged down by the presence of large files. We also discovered other problems. Printed reports—an important component because the program would have to produce not only finding aids but also the results of on-line searches—were difficult to set up and difficult to modify. The data structure could not be modified except by completely erasing the file and reentering the data. For no apparent reason, indexes became corrupt, and on several occasions hundreds of records were lost.

Because of these problems, I proposed dBASE III+ for archives use. Concerns were raised about dBASE's lack of compatibility with the MARC format and the feasibility of having two database management systems within Special Collections. I was assigned to investigate database management systems used in archival and manuscript repositories and make recommendations to the library. The head of Special Collections and associate dean for Collections and Services gave me permission to use dBASE on a trial basis, pending the outcome of my investigation.

Investigating the Options

The investigation of database management programs began in February 1990. The first step was ascertaining what programs were used in archival repositories. To find out, I queried archival colleagues in the St. Louis area. At this beginning stage, I was interested only in basic information: what programs were used, who the manufacturers were, how much the programs cost, and strengths and weaknesses

of the respective systems. Follow-up contacts were made with manufacturers, who provided sales literature, user- group information, and demonstration disks. Several additional programs came to my attention through a software review column in the *Midwestern Archivist.*[5]

The search resulted in a preliminary list of seventeen database management programs. Based on information from software reviews and comments from users, the initial field of seventeen was narrowed to seven: Advanced Revelation, dBASE III+, Georgetown Archives Management System, Marcon Plus, Workflow, MicroMARC:amc, and Minaret. The 1990 Society of American Archivists (SAA) meeting in Seattle played an important role in the database management project because it provided an opportunity to obtain detailed information about all seven programs. The Marcon, MicroMARC, and Minaret user groups would be meeting, and information-sharing sessions (called "swap shops") for users of Advanced Revelation, dBASE, and Minaret were part of the program. In preparation for the Seattle meeting, I reviewed the literature and user comments I had received for our final group of seven programs and worked out, in consultation with other staff in Special Collections, the criteria to be used for selecting our database management system. They were as follows:

- *Reliability.* Had other users experienced loss of data or system crashes while using a program?
- *Ease of use.* Factors to be considered included ease of installation and setup, amount of time and level of technical knowledge needed to learn the program, the extent to which the program would allow modifications in either the data structure or the data itself, and the quality of the user interface. A related factor was that no one on the Special Collections staff, and few people within the Olin Library System, had expertise in computer programming. Because of this, it was important that our database management system not be dependent on such expertise.
- *Adaptability.* The system should be adaptable to the needs of both the University Archives and the Manuscripts Section. The primary concern was whether the program could accommodate both folder- and item-level description.
- *Quality of documentation.* Is it easy to understand? Does the program come with a tutorial, either print or on line? If so, how useful is it in learning the program? A related factor was availability of resources beyond those provided by the manufacturer: are there user groups, classes, or books available to assist the user?
- *Manufacturer's support of the product.* Does the customer have to pay for technical support? How much support, if any, is included in the purchase price, and what is the cost of ongoing support? Do users have difficulty getting through to the manufacturer? Are they happy with the service they get? In the case of programs developed by an individual, how much technical support could we expect from the developer and how much would it cost?
- *Cost implications.* Cost was interpreted not only as the cost of the program itself but also as costs associated with technical support and the level of hardware needed to run the program. It was important to have a program that would run on our existing hard-

[5]Glen McAnich, ed., "Reviews: Computer Applications Programs," *Midwestern Archivist* 11, no. 1 (1986): 69–83. The programs reviewed were dBASE III, PFS File/PFS Report, DataEase, Savvy PC 4.0 and 5.1, Marcon II, PC File III, and DB Master 4 Plus.

ware with little or no sacrifice of performance. Could the program run on our local area network?

Comments from the Marcon Users Group at the Seattle meeting confirmed what I had experienced. Others had experienced problems such as sudden and unexplained locking of the keyboard, corrupted indexes, data loss, and report forms that produced garbage text. The comments related to performance alone were enough to remove Marcon from contention, but the users had other concerns: a poorly written manual, lack of a tutorial, and poor technical support from the manufacturer.

Marcon's manufacturer, Interactive Support Services, did not send a representative to the meeting, with the result that the chair of the user group had the unenviable task of addressing the concerns of a hostile group of Marcon users. The manufacturer was working on a new release of Marcon that would fix the many bugs in the program, but the release had no definite shipping date. It was also announced that all work on the development of MARC-MARCON, a Marcon Plus utility that would have given the program the capacity to create MARC records, was being abandoned because the archival market was too small to warrant the costs involved. By the end of the meeting, many user-group members were speaking openly about plans to abandon Marcon. Their comments made it plain that we, too, would be best served by moving in a new direction. Fortunately, we were in a position to do so because our investment in Marcon had been small.

Workflow and the Georgetown Archives Management System (GAMS) are derived from dBASE III+. Workflow is written in the dBASE programming language; GAMS is written using the dBASE language and the Clipper compiler.[6] Both were designed to meet the needs of specific institutions (UCLA and Georgetown University, respectively) by staff from those institutions.

The Georgetown system recognizes three levels of hierarchical description used in archives and manuscripts: collection, box, and folder. Data for each level is linked to the next with a machine-generated ID number. Index terms, filled in by the user during data entry, may be linked to each folder record and can be searched. The results of searches can be displayed on screen or printed, and the system can generate finding aids at folder level.

In creating the Workflow system, the developers began with the premise that processing requires a number of products above and beyond the finding aid, such as gift acknowledgements, monthly and annual statistics, inventories of archival supplies, and, of course, the MARC record.[7] Workflow consists of a series of databases and programs which are designed to track the actions taken on a collection, beginning with the initial contact with the donor and continuing with accessioning, creating a finding aid, and cataloging. Information pertaining to a given collection exists independently in the various databases until the programs format the data into whatever product is desired: accession register, gift acknowledgement (including news releases), finding aid, or MARC record, as appropriate.

[6]A compiler is a program that converts a user's program files to stand-alone applications that do not require the presence of a particular program to run and can be legally distributed. The presence of Clipper means that GAMS, unlike Workflow, does not require the presence of dBASE to run. In fact, GAMS makes use of features found only in Clipper that prevent it from running directly in dBASE III+ or dBASE IV. One such feature allows GAMS to accommodate up to 64 kilobytes of free-text description per folder, bypassing dBASE's limit of 254 characters.

[7]For a detailed outline of the Workflow system, see Dan Luckenbill, "Using dBASE III+ for Finding Aids and a Manuscripts Processing Workflow," *Rare Book and Manuscript Librarianship* 15, no. 1 (1990): 23–31.

For both programs, my primary concern was adaptability: neither program was able to accommodate the item-level description needed by our manuscripts curator. Another concern was the availability of technical support. Ashton-Tate, then the manufacturer of dBASE, had a policy of not providing assistance to users of customized dBASE applications. The developers of Workflow and GAMS were full-time archivists in Los Angeles and Washington, D.C., respectively. The distance to St. Louis from either location would make site visits prohibitively expensive and difficult to arrange. Telephone service would have to be scheduled to accommodate the developers' work schedule.

Advanced Revelation (A-Rev.), manufactured by Revelation Technologies, was the most powerful program I saw. A- Rev. consists of an array of programming tools that allow the user to custom design a complete database management system without having to write programming code. These tools allow the developer to paint data entry screens (fields can be placed anywhere on screen, and the developer can determine how much or how little data shows on screen), develop multiple levels of menus, develop pop-up windows that provide the user with lists of options at any point, and employ multiple levels of data verification. Data fields are stored in a central data dictionary, allowing changes to be made in the database structure without requiring that the developer restructure the entire data file or modify an entire application. A-Rev. allows variable- length description and can accommodate records up to 64 kilobytes in size. Boolean and proximity searches are both possible; report forms used for finding aids can be developed and stored in a centralized reports library. If the tools provided are not adequate, the developer can create others, thanks to the presence of a programming language, R-BASIC, and an internal compiler and debugger.

All the A-Rev. users I met commented that the user pays a price for A-Rev.'s power in the form of a steep learning curve—the program is difficult to learn. Having read the program's sales literature, I had to agree with their assessment. Clearly, the program's power was going to present a significant obstacle for us. There were no A-Rev. user groups in the St. Louis area. No one in Olin Library had heard of the program, much less knew how to use it. Thus, we would be faced with mastering a difficult program with few local resources to draw on. Although the training issues were significant, even more significant was the discovery, gained from conversations with other A-Rev. users, of two significant hardware limitations. The first was that A- Rev.'s file management system could not handle volumes of data larger than 32 megabytes, thereby putting an upper limit on the amount of information we could store in our computers. The second was that A-Rev. would not run on our local network without significant reconfiguration of all our existing hardware. For those two reasons, A-Rev. was not considered the best option.

The database management program that emerged as the best option for Special Collections use was dBASE III+. Its cost was the lowest,[8] and its performance was the least affected by having to run on older, slower computers. Unlike A-Rev., dBASE III+ could easily run within our network and it set no limits on how much data

[8]The low cost was due in part to the fact that dBASE III+ was beginning to give way to dBASE IV. Although dBASE IV was the newer product, I never considered it for our use because the early versions of dBASE IV received poor reviews in the popular personal computer journals. dBASE III+, on the other hand, was a product with a proven track record, and Ashton-Tate had no plans to stop supporting it. Since that time, Ashton-Tate has been taken over by Borland International, and dBASE III+ is no longer manufactured or supported. Borland has made a number of improvements to dBASE IV, and we will be upgrading our database manager in the near future.

could be stored—dBASE III+ can handle as much as the computer's hard disk can hold.[9] Unlike GAMS and Workflow, dBASE III+ could accommodate the differing descriptive needs of the University Archives and the Manuscripts Division. Because dBASE III+ was a commercial product, rather than the product of an individual developer, customer service was a phone call away at any time. The software was widely used within the archival community, and the many tutorial books, reference guides, and third-party utilities designed for it constituted a virtual dBASE industry.[10] It had the additional advantage of strong institutional support. During the course of the database management study, the library administration had selected dBASE as the officially supported database manager. This meant that on- site service (if needed) and upgrades could be easily obtained. It also meant that, in terms of information sharing with other units, dBASE would not isolate us from the rest of the library system.[11]

One thing dBASE III+ could not give us was the ability to create MARC records that could be loaded into the OCLC database and our local NOTIS catalog. To that end, Minaret and MicroMARC were explored. We were evaluating not only the usefulness of these programs for creating and exporting MARC records, but also whether one of these programs would substitute for, or serve as an adjunct to, dBASE III+. I preferred Minaret to MicroMARC because it had a more user-friendly interface, could work with word-processing programs such as WordPerfect to create finding aids and catalog cards, could read dBASE files, and was more widely used by archival colleagues in the St. Louis area.

While in Seattle, I spoke with representatives from the manufacturer and attended the Minaret users group meeting. On returning to St. Louis, I obtained demonstration disks, tried out the program, and consulted with Minaret users in the St. Louis area. In the end, we decided against a PC-based MARC AMC utility and in favor of a modem with OCLC's Passport software.[12] Besides being a more cost-effective solution, Passport would allow us to enter MARC AMC records directly into OCLC without having to convert data into a format OCLC could read, as would be necessary with Minaret or MicroMARC.[13] Another benefit would be access to the OCLC authority file and other utilities we would need for our cataloging.

[9]dBASE III+ has a limit of one million records *per file* but no limit on the number of files that can be created. In effect, dBASE can handle as much data as can fit on the hard disk. With Advanced Revelation, 32 megabytes is all the program can handle, even if the hard disk has 200 megabytes of free space. This problem can be solved using multiple DOS partitions, but such partitioning is not possible with later versions of DOS.

[10]A related consideration at the time was that both Ashton-Tate and dBASE had remained stable for many years; thus we could be reasonably confident that Ashton-Tate and dBASE would be stable entities over the long term. Within six months after the conclusion of the database management study, Ashton-Tate became a subsidiary of Borland International, one of dBASE's former competitors.

[11]It should be noted that the Special Collections Department was not at any time forced to use dBASE. The library administration encouraged us to look at a number of options and propose the solution we felt was best.

[12]Passport is the terminal emulation software for OCLC's PRISM system. In layman's terms, Passport enables a personal computer to function as an OCLC terminal.

[13]Minaret could send records to OCLC over telephone lines using the ProComm telecommunications package and a third-party utility; however, this required a number of data conversions. For a discussion of Minaret's uploading procedure, see Carson, "American Medical Association." MicroMARC had, at the time, no way to send AMC records to OCLC via telephone lines. MicroMARC users had to copy completed records to a floppy disk and send them to Michigan State University. At Michigan State, records were tape-loaded into OCLC via the university's mainframe. Both MicroMARC and Minaret have since added modules for importing and exporting MARC records.

The study of database management needs in Special Collections ended in October 1990, with a two-part recommendation to the library administration: Full access to OCLC and NOTIS for cataloging needs, and dBASE III+ for in-house database management functions, including the preparation of the finding aids that make a MARC record possible. This recommendation was accepted, and work with dBASE began in November 1990.

Implementing the System

Once dBASE was installed, the next tasks were staff training and putting dBASE to use in the archives. The Assembly Series database, created during the initial trials with dBASE and originally designed with that specific collection in mind, was modified so that it could accommodate audiovisual materials from all collections. Databases for folder-level description of paper records and item-level description of printed items were added. Because our databases are grouped along lines of format (paper, audiovisual, or print), it is possible for items from the same collection to appear in three different databases. To keep track of where information was stored, and to assign classification numbers, two more databases were created for collection-level data—one for university records and one for our St. Louis–area manuscripts.

The implementation of dBASE was accompanied by changes in our processing procedures for paper records. Emphasis was placed on folder-level, rather than item- level, processing of paper records. This change in emphasis maintains intellectual control over collections while reducing the amount of staff and student time needed to process them. Another change is that we no longer create draft finding aids at various stages of processing. As part of the arrangement and description process, we review old folder headings and assign new ones when appropriate, as we did in the past. Once the arrangement and description are complete, the processor, working from the folders themselves, enters the finished folder-level data into the computer database. When all folders have been entered, the processor generates the finding aid using the dBASE report form. Audiovisual materials and printed items are described at item level, but they are checked in using the computer rather than manually. The computer greatly simplifies the process of shelving an item, updating box and folder numbering, and producing a corrected finding aid.

Of course, dBASE could accomplish nothing unless the staff and student assistants were trained to use it. Like all university archivists, I am faced with a constant turnover of student assistants, so dBASE training is ongoing. It is also incremental. In teaching a student the basics of dBASE, I begin with an introduction to our databases and the types of materials they describe. This introduction serves a dual purpose: in explaining what the various database fields mean, I am also giving a primer on the principles of archival arrangement. Because the manuscripts, publications, and audiovisual databases have similar data structures, learning to navigate one file means that the others can be quickly mastered. The students are first taught the four most basic commands used in data entry: USE, EDIT, APPEND, and QUIT. Once those commands are mastered, I move on to searching commands, such as LOCATE, LIST, and DISPLAY, and global-replace commands such as REPLACE WITH that expedite data entry by reducing the amount of repetitive typing. With few exceptions, students are comfortable with the computer and have little trouble with dBASE commands.

Once the students are familiar with both dBASE and basic archival hierarchy, they are introduced to actual processing of collections. For them, processing is a welcome addition to the more routine tasks of refoldering, paging and retrieval, and pho-

tocopying. For me there is the challenge of assigning appropriate work. As a recent *American Archivist* article rightly points out,[14] student assistants cannot substitute for staff and their work assignments must reflect that fact. When I assign processing projects, the students are given collections that need little rearrangement but do need review of folder headings and input into the computer. Collections that require complex rearrangement, require access decisions, or have significant preservation problems are processed by my assistant. The staff consists of one half-time paraprofessional and two undergraduate student assistants, and it is not unusual to have three or four projects in progress simultaneously. The volume of materials processed with the same number of staff has increased dramatically.

MARC records, however, are not a student task. Because I was to be responsible for creating catalog records for archival collections, I, too, needed training, some of which was provided by technical services staff in Olin Library. Like many archivists, I have taken the SAA workshop on *Archives, Personal Papers, and Manuscripts*; I have also taken a course on OCLC authority files sponsored by our regional OCLC office. Additional workshops will be necessary in order to keep up with current cataloging practice.

The procedures used for creating MARC records take full advantage of OCLC's ability to copy screens to a personal computer's hard disk. First the OCLC workform for AMC records is copied to the computer's hard disk and saved as a WordPerfect file. This procedure allows extensive editing of the record without incurring large amounts of connect time. Once the basic information (main entry, title, physical description, organization and arrangement, restrictions, scope and content, biographical/historical note) is in final form, the WordPerfect file containing the filled-in workform is printed out. Data from the printout is keyed into OCLC and added to the OCLC save file. Potential subject and added entries are searched in the OCLC authority file and appropriate entries are added to the saved record. Technical services staff review the completed record for conformity with both OCLC conventions and AACR2, then the record is added to the OCLC database. OCLC records are loaded into the library's NOTIS system via the library's weekly tape load, without the need for further intervention on our part.

As of February 1994, our databases contain approximately 25,000 collection-level, folder-level, and item-level records spanning over 70 record groups. As the databases grow in size and scope we gain an increasingly useful on-line searching tool. Already we have gained greater productivity from the same number of staff. Now that the various components of our automation program are up and running, keeping it running smoothly is an important activity. To that end, we perform regular backups of our data and regular checks of our hardware for viruses and signs of impending hard-disk failure. Planning for the future is already taking place in a number of areas. A number of special collections departments are now using the Internet communications protocol known as Gopher to make finding aids available over the Internet. We are exploring ways to do the same.[15]

Planning and developing an automation program taught me two important lessons. The first was that developing an automa-

[14]Barbara L. Floyd and Richard W. Oram, "Learning by Doing: Undergraduates as Employees in Archives," *American Archivist* 55 (Summer 1992): 440–52.

[15]Seventeen special collections departments have set up Gopher servers as of February 1994, and the number continues to increase. An important part of making our finding aids accessible over Gopher will be retrospective conversion of older finding aids that exist only in typed form.

tion program has both technical and managerial components, and neither is more important than the other. In fact, the two are constantly overlapping. The second lesson was that an automation program is a continually developing process. In the beginning I assumed that selecting a system would mean the end of my work. I now know it was only the beginning of an ongoing, long-term process.