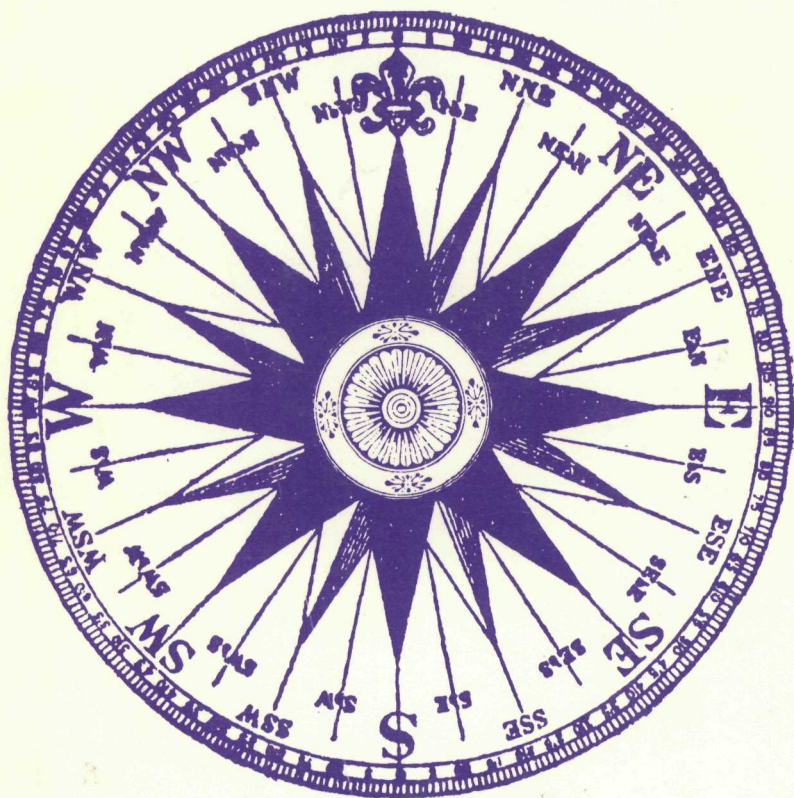


The American Archivist

Volume 51
Numbers 1 & 2
Winter and Spring 1988



Published Quarterly by
The Society of American Archivists

The American Archivist

Julia Marks Young, *Editor*

University of Southern Mississippi

David Klaassen, *Editor as of 1989*

University of Minnesota

Bill Burck, *Managing Editor*

Phyllis Pittman, *Copy Editor*

DEPARTMENT EDITORS

Dean DeBolt and Joel Wurl, *Commentaries and Case Studies*

Marjorie Barritt and Nancy Bartlett, *The International Scene*

Glen Gildemeister, *Reviews*

Bill Burck, *From the Archives*

EDITORIAL BOARD

Lewis J. Bellardo (1987–1991), *Georgia Historical Society*

Edmund Berkeley, Jr. (1985–1989), *University of Virginia*

John Daly (1987–1991), *Illinois State Archives*

Kenneth W. Duckett (1984–1988), *University of Oregon*

Jacqueline Goggin (1985–1988), *Library of Congress*

Susan Grigg, chair (1986–1990), *Smith College*

Jane Nokes (1987–1990), *Bank of Nova Scotia*

Nancy A. Sahli (1985–1989), *National Historical Publications and Records Commission*

The Society of American Archivists

PRESIDENT Sue E. Holbert, *Minnesota Historical Society*

VICE PRESIDENT Frank B. Evans, *National Archives and Records Administration*

TREASURER Anne P. Diffendal, *National Museum of Roller Skating*

EXECUTIVE DIRECTOR Donn C. Neal

COUNCIL MEMBERS

Paul I. Chestnut (1985–1988), *Library of Congress*

Richard J. Cox (1986–1989), *New York State Archives and Records Administration*

Maygene Daniels (1987–1990), *National Gallery of Art*

Linda Edgerly (1986–1989), *Consultant*

Anne R. Kenney (1985–1988), *Cornell University*

Archie Motley (1987–1990), *Chicago Historical Society*

Mary Jo Pugh (1987–1990), *Consultant*

Victoria Irons Walch (1985–1988), *Consultant*

Joan Warnow-Blewett (1986–1989), *American Institute of Physics*

Cover note: This illustration is from *Planning for the Archival Profession: A Report of the SAA Task Force on Goals and Priorities*, which outlined specific goals and objectives for the profession. Several of the articles and commentaries in this issue relate to one of the three principal goal areas in the GAP Report.

The American Archivist

Volume 51 / Numbers 1 and 2 / Winter and Spring 1988

Archival Education: Two Fables / 16

William L. Joyce

Introduction to Archival Research Agendas / 24

Charles G. Palm

The Archivist's First Responsibility: A Research Agenda to Improve the Identification and Retention of Records of Enduring Value / 28

Richard J. Cox and Helen W. Samuels

Commentary / 43

Frank Boles

Commentary / 47

Frank G. Burke

The Management of Archives: A Research Agenda / 52

Paul H. McCarthy

Commentary Wilsted / 70

Thomas Wilsted

The Role of Use in Defining Archival Practice and Principles: A Research Agenda for the Availability and Use of Records / 74

Lawrence Dowler

Commentary / 87

Jacqueline Goggin

Commentary / 91

Anne R. Kenney

Commentaries and Case Studies / 98

The NHPRC Data Base Project: Building the "Interstate Highway System" / 98

Richard A. Noble

Who are the Archivists and What Do They Do? / 106

Maynard Brichford

A Positive Approach to Negatives: Preserving Photographs via Microfilm Technology / 111

Ruth B. Kerns

Format for Cooperation: Cooperative Collection Registers at the Peabody Museum of Salem and the Essex Institute / 115

Robert P. Spindler, Gregor Trinkaus-Randall, and Prudence Backman

The International Scene / 120**Archives in Emerging Nations: The Anglophone Experience / 120**

Philip Alexander and Elizabeth Pessek

Archivaria: A Brief Introduction to the Journal of the Association of Canadian Archivists / 132

Terry Thompson

Luncheon Remarks to the Society of American Archivists, 4 September 1987, New York / 135

Charles Kecskeméti

Luncheon Remarks to the Society of American Archivists, 4 September 1987, New York / 137

Margarita Vazquez de Parga

International Council on Archives, Executive Committee Annual Meeting and the International Round Table Conference on Archives, September 1987 / 139

Robert M. Warner

The Forum / 6

Reviews / 142

The Society of American Archivists / 162

The Annual Index, 1987 / 185

REVIEWS**Walch, comp., *Our Family, Our Town: Essays on Family and Local History Sources in the National Archives* / 142**

reviewed by Mark H. Jones

Couture and Rousseau; Homel, trans., *The Life of a Document: A Global Approach to Archives and Records Management* / 143

reviewed by Ann Bowers

Howe, Fleming, Kemp, and Overbeek, *Houses and Homes: Exploring Their History* / 145

reviewed by John M. Bryan

Howe and Kemp, *Public History: An Introduction* / 146

reviewed by Patrick B. Nolan

Sellers, comp., *Civil War Manuscripts: A Guide to Collections in the Manuscript Division of the Library of Congress* / 148

reviewed by Louis H. Manarin

Kline, *A Guide to Documentary Editing* / 149

reviewed by Mary Lynn McCree Bryan

Wadlow, *The Disposition of Government Records*; McGee, comp., *Records Management Resources: An Annotated Bibliography for Local Governments* / 151

reviewed by Gerald G. Newborg

DeGruson, Kelly, Mundy, and Quiring, eds. and comps., *A Guide to Special Collections in Kansas*; Department for Libraries and Archives, *The Guide to Kentucky Archival and Manuscript Repositories* / 152

reviewed by Richard A. Cameron

Chadwyck-Healy, *National Inventory of Documentary Sources in the United States. Part 3: State Archives, Libraries and Historical Societies and Part 4: Academic Libraries and Other Repositories* / 154

reviewed by Leon J. Stout

The American Archivist is published by the Society of American Archivists, 600 S. Federal, Suite 504, Chicago, Illinois 60605, four times yearly. Postmaster: send address changes to Society of American Archivists, 600 S. Federal, Suite 504, Chicago, Illinois 60605. Subscriptions: \$30 a year to North American addresses, \$35 a year to other addresses. Single copies: \$8 to SAA members, \$9 to nonmembers.

Articles and related communications should be sent to Bill Burck, Managing Editor, Society of American Archivists, 600 S. Federal, Suite 504, Chicago, Illinois 60605. Telephone: (312) 922-0140. Advertising correspondence, membership and subscription correspondence, and orders for back issues should be sent to SAA at the address above. Claims for issues not received must be received by SAA headquarters within four months of issue publication date for domestic subscribers and within six months for international subscribers.

The American Archivist is available on 16 mm microfilm, 35 mm microfilm, and 105 mm microfiche from University Microfilms International, 300 N. Zeeb Road, Ann Arbor, MI 48106-1346. When an issue is out of stock, article and issue photocopies may also be obtained from UMI. *The American Archivist* is indexed in *Library Literature* and is abstracted in *Historical Abstracts*; book reviews are indexed in *Book Review Index*.

Typesetting and printing of the *American Archivist* is done by Imperial Printing Company of St. Joseph, Michigan. *The American Archivist* is printed on an alkaline, acid-free printing paper manufactured with no groundwood pulp. As such, it substantially meets the requirements of the American National Standard for Information Sciences—Permanence of Paper for Printed Library Materials, ANSI Z39.48-1984.

The American Archivist and the Society of American Archivists assume no responsibility for statements made by contributors.

©The Society of American Archivists 1988. All rights reserved. ISSN 0360-9081. Second class postage paid at Chicago, Illinois, and additional mailing office.

The Forum

TO THE EDITOR:

I want to thank the *American Archivist* for publishing Jon Reynolds's sound review of *MicroMARC:amc* in the Spring 1987 issue. One point of clarification is that there is no records management module in development. The records management functions of the system are already incorporated in all the modules, but particularly in the report module. If anyone is concerned about the "compatibility" of their personal computer with *MicroMARC:amc*, we have a list of computers on which it has been tested and we will be happy to send a copy to anyone who requests it.

FREDERICK L. HONHART
Michigan State University

TO THE EDITOR:

As one of the co-chairmen of the Working Group on Descriptive Standards which published the report *Toward Descriptive Standards* reviewed in this journal (Spring 1987) by Richard Szary, I wish to offer a few comments.

Szary's thoughtful review touches on a number of very important issues, but, I think, he misses the mark in some of his judgments of the report. The Working Group was set up primarily to study the extent to which current descriptive practices in Canada might be standardized and to make recommendations about advancing sys-

tematization of those practices. We discovered that Canadian archives regularly capture the same data about holdings and reveal it in roughly the same instruments (no surprise), but they rarely systematize practices by developing in-house standards or using ones developed externally. Yet everyone seems to be crying for standards. In response to the great gap we saw between the expression of need and the more or less naturalistic habits of the profession (the reasons for which it seemed idle to speculate about), the Working Group opted to pare theorizing to the bone and offer a set of recommendations designed to guide Canadian archivists to develop the standards they so evidently needed. We urged archivists to apply existing standards where they had been developed and seemed appropriate to activities well established in the field, and we fashioned a series of recommendations to bring about some structured and professionally directed forum for further interchange and follow-up on the recommendations, which I am pleased to say has occurred within the structure of the Bureau of Canadian Archivists, in whose name the report was written. We now at least have a means, as David Bearman puts it, to agree to agree, and what is more, to act.

I quite agree with Szary that the report does not tie up all the loose ends of our practices in a neat package "to develop a focus and a discipline that can guide development of effective and integrated in-

formation systems for historical information,” surely a tall order, but I still think that archivists might profit from reading the report with the more practical object in mind of discovering the range of descriptive and related activities which can be standardized and the battery of existing standards which might be applied to their descriptive and indexing practices.

Copies are still available from me at the School of Library, Archival, and Information Studies, University of British Columbia, #831-1956 Main Mall, Vancouver, B.C., Canada, V6T 1Y3, free of charge.

TERRY EASTWOOD
University of British Columbia

REVIEWER'S RESPONSE:

Terry Eastwood is correct in asserting that my review of *Toward Descriptive Standards* addressed broader theoretical issues than those considered by the report's authors. Even more important than the question of which standard to use for a particular aspect of descriptive practice, are concerns about how we, as a profession, manage the standards process—deciding where standards are needed; evaluating proposed standards; distributing, promoting, and implementing agreed-upon standards; and monitoring compliance with them. The report does address these issues, and the follow up by the Bureau of Canadian Archivists on these recommendations with a “structured and professionally directed forum” to guide further efforts and discussions is perhaps the most significant and heartening development following from the report—a development that U.S. archivists are only now beginning to work towards.

The question of practical vs. theoretical approaches to standards, raised by Eastwood, is a legitimate concern. Using current practice as a basis for standards development, as the Working Group suggests, has the advantage of building on a

more-or-less common base of knowledge, and offers the hope of a timely return on the development effort. What is dangerous, as Walt Crawford points out in a general discussion of the standards process in *Technical Standards*, is the exclusive reliance and enshrinement of current practice as the only source for standards development. While I doubt that Eastwood or other members of the Working Group would claim that this was their intention, some of their recommendations come very close to doing just that—elevating particular applications of a general principle into the principle itself. My hope is that the follow-up efforts would strive for solutions that will accommodate current practice without closing off possibilities for further development.

My intention in the review was to treat this report as a “work in progress”—the first comprehensive review of the state of descriptive standards in the profession. As such, I would consider it an admirable effort to synthesize and focus professional concern on an increasingly vital area of archival practice. While not a flawless blueprint from which we can begin the detailed construction of archival standards, it provides us a working drawing from which we can now begin to discuss and debate how to proceed in a thoughtful yet practical process of definition and implementation.

RICHARD SZARY
Yale University

TO THE EDITOR:

Ronald Zboray's article in the Spring 1987 *American Archivist* is interesting for its description of how one determined user made his copy of dBase III stand on its head and sing *The Star-Spangled Banner*. Aside from that it is confused, confusing, and misleading, except insofar as it may serve to convince readers that most PC data base management systems are ill-suited to the task of managing MARC-like data. That

fact should not be surprising: the PC DBMS market consists principally of homes, shops, and offices whose applications involve relatively simple data lacking the three structural features which characterize real MARC records—repeatable fields, extreme variability of field length, and potentially very long records.

My two main quarrels with the article concern its perspective and its misinformation about the AMC format. Zboray's view of the world is dBASE-centric; that is to say, he starts from the existence of this product and looks outward towards an application which, in fact, he never actually describes. (After an automation session at SAA some years ago, a woman asked me, quite simply, "What kind of micro-computer should I buy?" I'm afraid my equally simple response—"What do you want to do with it?"—left her quite speechless.) Data entry, storage, and a vague reference or two to searching do not describe the functions Zboray's system is intended to provide or the activities in a repository it is meant to support. He does make passing mention of a reporting capability; presumably that will be able periodically to print purchase orders for additional hard disks.

When it comes to MARC AMC, the article helps to perpetuate one of the most common misunderstandings about the format, namely the absolute distinction between, on the one hand, the USMARC Formats for Bibliographic Data, themselves a national implementation of an international standard, and the AMC format as a particular elaboration of those formats, and, on the other, any implementation of a system which uses data conforming to those standards. Let me be more specific.

Zboray's dBASE implementation uses fields which largely parallel those defined in official AMC documentation; how he handles subfielding is not at all clear—the statement on page 220 that "a menu-driven program itself generates this type of input

[subfield delimiters] without the assistance of the user" is either a printer's error or requires a great deal of explanation. What he does *not* say is that he can import or export a MARC record which conforms both structurally and in terms of content designation to the defined format. He does not tell us that his system can read in a record in the standard exchange format and unpack it into his internal processing format. Nor does he describe how he would go about generating an exchange format record—does his system have the capability of building a record directory, and how would he handle the quite long data objects which could result from assembling the rather ingenious packets described in his final model?

Having found a way to deal reasonably efficiently with the problem of variable-length fields in a fixed-length environment (from the storage aspect only, of course; the one-data-base-per-field strategy requires considerable processing to assemble a record for display or transmission), Zboray is still unforthcoming on the repeatability question. He finesses the issue for the 5XXs by lumping them all into memo fields, but in doing so he loses the ability to index the powerful 541 and 583 fields; maybe he doesn't want to retrieve by donor or to check the expiry dates of access restrictions. But I cannot figure out how he proposes to have several dozen personal and corporate name subject headings in a single logical record, an extremely common situation for MARC AMC users.

Again, the confusion between definition and implementation is obvious when he states, referring to a dBASE restriction, that "[a] maximum of 128 fields promises to fit easily MARC AMC's requirement for 77 fields," and when he adds shortly thereafter, "The MARC AMC format itself encourages a waste of disk storage space." Watch my lips: *The MARC format* is silent regarding disk space; *the MARC format* does not require 77 fields.

If a system is designed to store records

in the MARC format, as opposed to storing data elements which are semantically equivalent to those defined in the format, then its use of disk space will be extremely efficient; there is the overhead of the record directory, but the variable fields occupy only as much space as there is data, no padding, no unusual packaging. Most systems do not store records in that form, however, since it is not particularly efficient for real-time processing. Instead, the format is used for its intended purpose: communication within and between systems.

The number of fields in a given record is a function of the process of description the record is intended to reflect. The format defines which fields are *valid*; the descriptive rules and conventions, e.g., Hensen, prescribe how certain kinds of information should be identified, assembled, or constructed to describe given materials; and examples and compendia of practice, e.g., LC's MFB examples and Evans and Weber's MARC, bring the two pieces together. In the AMC data base with which I am most familiar, there are records with eight or nine fields and records with two hundred or even three hundred fields. There is not a single record—there cannot validly be a single record—that contains the 77 defined fields. To give just one example, fields 100, 110 and 111 are mutually exclusive. (I don't know if 77 is indeed the number of defined fields, or just the ones Zboray has chosen to implement. Since, for instance, he throws out 066, mistakenly referring to it as 006, but retains 880—the two really do need to go together, but Zboray doesn't appear to understand the use of either, saying he excluded “006” “because dBASE stores data in ASCII format”—I didn't bother to correlate his listing of fields with other documentation.) In reality, it is dBASE itself that requires those 77 fields, because it needs to allocate space in each record for every data element defined for the record, regardless of whether or not an element has a value in a given

record. Not every data base manager does that, not every PC data base manager does that, and Zboray knows that.

The need for PC-based systems to support processing in archives and manuscript repositories, whether stand-alone or linked to and complementing larger shared systems, is undeniable and, as Zboray rightly notes, as yet not satisfactorily filled. The principal shortcoming of most of the PC AMC implementations that exist today is that supporting the format, whatever that means, has too frequently been seen as the primary design goal, the most significant selling point, or both. Were I in the market for such a system, I would certainly want one which would allow the creation of records semantically compatible with MARC AMC, and which was capable of importing and exporting MARC data in exchange format. But the overall functionality of the system—what it can do with the data—is a much more central issue. Form, and in this case format, follows function in systems design, or at least it should. Ronald Zboray's four approaches take a very different direction.

ALAN TUCKER

Research Libraries Group

TO THE EDITOR:

Since 1983 we have worked on the programming for the Michigan State University stand-alone microcomputer system, *MicroMARC:amc*. During this time we have become very familiar with the USMARC AMC format and its implementation on microcomputers. We have over twenty years experience with mainframes and minicomputers, data base systems, statistical packages, and other computer applications.

In the spring 1987 issue of the *American Archivist* there was an article “dBASE III Plus and the MARC AMC Format: Problems and Possibilities” by Ronald J. Zboray. This article describes four methods of using dBASE III Plus to enter, search, and report on MARC AMC records.

All of the methods proposed in the article fell short of actually producing a record in the USMARC AMC format. To understand why this is true, let's discuss what the USMARC format is. The USMARC format consists of three parts: the leader, the directory, and the MARC fields.

The leader is used to store certain information about the structure and contents of each USMARC record. Some bytes of the leader are the same for all AMC records, but there are a few leader bytes which can be different. These bytes need to be entered by the user. The leader also contains the total length of the record and the byte where the MARC fields begin. These two numbers can only be determined once the entire record has been processed and its directory has been built.

The directory is a sorted list containing the tag of each MARC field stored in the record, its position in the record, and the number of characters it contains. It is necessary to examine the entire record when the directory is being built.

Finally, the actual bibliographic information is stored in the MARC fields. Each MARC field (except the 00X fields) started with a two-byte indicator code. The information within each field is often further divided into subfields. In addition, if more than one set of information is available for a single MARC field, that field can be repeated.

The dBASE III data bases described in the article include the MARC fields, but not the leader or the directory. This means that these records are *not* stored in the USMARC format. No mention is made in the article of how to convert a dBASE III data base to the USMARC format. Nor is any mention made of how to add a USMARC formatted file to a dBASE III data base. In our opinion, it would not be easy to accomplish either of these conversions using dBASE III.

There are also several features of the USMARC format which are not discussed in the article. No mention is made in Table 1

or elsewhere about entering the needed leader bytes. Indicators and repeatable fields are also not discussed.

After reviewing the article, we do not believe that any of the four methods are simple enough for an average computer novice to install without professional aid. These individuals would probably need extensive training in how to use dBASE to do searches to run reports as well as enter records. The third and fourth methods are so complicated that we believe that it would be necessary for a programmer to be kept on staff if either of these methods were used.

The following sections are our comments on each of the four methods.

Method I. Method I describes the basic method by which data bases are stored using dBase III Plus. The main problem with the basic method is that it is very wasteful of disk space. Each field must contain enough bytes to fit in the largest possible entry for that field. This means that even when a field contains information, there are still many wasted bytes. In addition, many of the fields are only used in some of the records. But the space for unused fields is still needed by the data base. As stated in the article, only about half of the space used by the data base actually contains information. In addition, it is not possible to include all of the MARC tags in such a dBASE III data base. This necessitates the use of memo fields. The use of memo fields means that a separate file is created for every MARC record where the memo field is used.

Method II. Because method I had a maximum limit of about 5,000 records, this method was created to double the maximum number of records a data base could store. This is done by using two dependent data bases which are linked by a common field. This method would have about the same amount of wasted disk space as method I.

Method III. This method uses additional dBASE programming to eliminate some of the instances of wasted disk space. Unfortunately, this is done by making the

basic setup of the data base far more complicated. This change may also make doing searches and reports more complicated. Even though the programming makes these changes transparent to the user, its complexity would make it necessary to have a programmer on staff to keep the data base running correctly and aid the staff in making searches and creating new reports.

Method IV. This method makes the programming situation even more complex by setting up a separate file for each MARC tag. An indexing record is used to indicate which fields are present in each MARC record. Thus, space is used only for fields actually present in the record. However, space is still wasted for fields that are not completely filled with information. As with method III, a programmer on staff is necessary. This method also requires that dBASE III open and close each of its files whenever a record is loaded or stored. We estimate that with the number of files needed it would take about two to three minutes to load a record to the screen.

The goal of creating extremely large data bases is not practical. A file containing more than 100,000 MARC records would be too large to process in a reasonable amount of time with the current generation of microcomputers.

ALICE G. KALUSH
Michigan State University

ANDERS JOHANSON
Michigan State University

AUTHOR'S RESPONSE:

I never intended to pose the dBASE implementation described in my article as a competitive commercial data base program to draw business away from either RLG or MicroMARC:amc. Nor am I any sort of dBASE salesman in disguise.

The realities of microcomputer use in archives motivated me to write the article.

According to Lisa Weber's recent survey of archival automation in 261 repositories, "dBase in all its various iterations is by far the most popular microcomputer data base management package" (*SAA Newsletter*, November 1987, 4). More than three times as many archivists used dBASE as any other data base management system; the runners-up all were very similar to dBASE in their limitations. The ubiquity of dBASE among archivists is a fact which must be reckoned with. As the article's title, "dBASE III Plus and the MARC AMC Format: *Problems and Possibilities*," indicates, I wanted to consider the issue of dBASE compatibility with MARC AMC for the many archivists who use that particular data base management system. Throughout the article I speak repeatedly as much about the problems as their possible solution, and I make clear just how difficult it is to begin to adapt dBASE to MARC AMC compatibility.

The two letters also ignore an important distinction. I speak in the article of MARC AMC "compatibility," not, as they seem to assume, about creating full-blown MARC AMC records to be stored in a local data base ready at any time for uploading to a main system. One of my goals was to give archivists using dBASE some tips about the organization of the information in their fields so that they would not make mistakes in data base design which would prevent them from ever outputting their data in a MARC format. The idea of data transfer between dBASE and a MARC system would itself require its own, highly technical article. The question I addressed was: how can archivists store information *locally* in dBASE with the future possibility of uploading to a MARC AMC mainframe? I would like to think that my article either gave dBASE users some tips about dealing with MARC AMC, or that my clear caveats concerning disk space wastage and the programming it takes to get the most efficient system could be used as arguments against using dBASE at all.

Response to Tucker

In his first paragraph Tucker reveals a mainframe-user bias against the PC. PCs are not the orphan children of mainframes, or even minis; but have a life and integrity of their own. I suggest Tucker investigate more closely just how elaborate dBASE applications have become, particularly in business. Contrary to Tucker's statement, most dBASE applications are not found in homes or shops but in small to moderate-sized businesses. A combination of text-file-based dBASE "memo" fields and the use of the relational capability of the data base to extend field lengths is well known and well used in business applications and can cope with what Tucker sees as "the three structural features which characterize real MARC records—repeatable fields, extreme variability of field length, and potentially very long records."

Tucker seems to create a straw dog when he introduces the term, "Zboray's system." I never applied the word system to my discussion of dBASE. From this point on, Tucker discusses the "system" and attempts to evaluate it as a product available for mass distribution. Because there is no product, there is no export-import utility or directory-building program. Most of the users of dBASE to whom the article was addressed should be well aware of dBASE's capabilities for "data entry, storage, and searching." It was not within the purview of the article to go into its capabilities.

Tucker's statement concerning subfielding ignores the capabilities of menu-driven programs. A menu that prompts the user to enter something as simple as a date without a subfield delimiter is quite common in programming. In fact, if one boots up PC-DOS on a machine without an internal clock, one is asked to enter the date in the same fashion without a subfield delimiter. In a dBASE application, it would not be difficult to output the data immediately preceded by a two-character delimiting code.

Next Tucker criticizes my approach in the discussion of variable-length fields in a

fixed-length environment as requiring "considerable processing to assemble a record for display or transmission." This seems unrealistic, based upon my experience with such an approach as implemented at the Emma Goldman Papers. It takes less than three seconds—much faster than the operation of many modems—to paint an entire screen with text retrieved from a data base holding portions of fields.

I believe that I addressed and did not, as Tucker charges, finesse the field repeatability problem. Although dBASE III Plus itself does not yet provide for searching in memo fields, I mentioned in the article one inexpensive commercial utility that will. And the dBASE memo field, with the help of an external word processor, can be any length, so that it can incorporate several dozen personal and corporate names. Another approach, followed by the Goldman Papers, but too complex to discuss in the article, is to use a coding system for names in their own data base linked by record number or accession number to the main record; the data base of codes is itself linked to a data base where the authorized name forms appear. In this manner, any number of name-type subjects may be assigned and one authority record replaces all occurrences of the code when records are reported out.

I believe that Tucker makes much ado about very little concerning my suggestion that MARC AMC encourages dBASE users to waste disk space and that the seventy-seven basic field types in MARC AMC require a similar number of fields in dBASE. Tucker seems to have read my statements out of the context of a dBASE environment. Although he adequately describes the nature of the suggested approaches as "storing data elements which are semantically equivalent to those defined in the [MARC AMC] format," he pushes the point that the MARC format itself will have no extra "padding" in its variable-length fields—as if that was not common sense. As surely he must know, any dBASE ap-

plication will have a fixed number of named, defined fields (unless one starts entering customized subfield delimiters to pull out of larger dBASE fields for AMC fields that cannot possibly be used in the same record).

In short, Tucker seems to have gone through a great deal of indirection to argue against using not only dBASE but the system he fancies I have conjured up. The results of his arguments threaten to impugn the quality of the article, when Tucker's disagreement is with using dBASE for MARC AMC at all—a perspective to which I strained to lend credence.

To his credit, Tucker found one typographical error in which tag “066” was rendered as “006”—an error which crept in during typesetting and was missed by all proofreaders. And he was correct that I should have left out tag 880, although my point was not to dictate what tags should or should not be used, but rather to show that some tags are dispensable.

In my view, Tucker's peroration seems to show that he misunderstands the fundamental attraction of dBASE, for he accuses me of ignoring the overall functionality of a dBASE-based system. The large, well-established base of dBASE users among archivists and in business of all sorts seems itself an argument for the superior functionality of the system. If Tucker does not like the idea of using dBASE or any PC fixed-field based DBMS for MARC AMC, he should simply say so.

Response to Kalush and Johanson

Kalush and Johanson also seem to misunderstand the purpose of my article. The heart of their criticism is that I do not discuss creating full-blown MARC records stored in dBASE but rather only MARC *compatible* records. They particularly criticize me for not dealing with the leader and directory portions of MARC. I specifically decided not to engage in this type of technical discussion in light of Nancy Sahli's introduction to *MARC for Archives and*

Manuscripts: The AMC Format, to wit: “The leader and record directory, used in automated implementations, are primarily *machine-generated entries* [emphasis mine]. . . .Conceptually they're rather hard to understand and incorporation of them into a system design is best left to computer professionals.” From my long experience in programming dBASE, I can safely say that a program which would generate this information would not be very difficult to write; however, such preparation for data transfer between dBASE and a MARC system was not the topic of my article.

Although Kalush and Johanson claim, in their review of the four methods, that novice users of dBASE would not be able to install approach I or II, they never argue their case, but simply pronounce that “professional aid” would be necessary. Creating a data base in dBASE with the aid of the program's user-friendly “assistant” is not very difficult; one need only name fields, determine their type (whether character, numeric, or memo, for example), and enter the maximum number of characters the fields will contain (field length). Entering information into records does not require “extensive training in how to use dBASE.” One simply enters “USE {filename}” and then “APPEND”; a blank record appears on the screen with the fields presented for data entry. Ctrl-W brings the user back to the dot prompt. Running searches and reports, for the most part, is similar to other types of automated searching which employ Boolean logical operators—the basic principles do not take very long for anyone to learn. As for approach II, I do not claim that novice users (not novice programmers) would have an easy time setting up dependency but rather that “the limited programming involved can be accomplished by anyone with a *working knowledge* of dBASE” (emphasis added). And I argue that approach II still remains in light of more sophisticated applications a reasonable, cost-efficient alternative, even “if outside programmers must be hired to

set up the dependency.” Kalush and Johanson also claim “the third and fourth methods are so complicated that we believe that it would be necessary for a programmer to be kept on the staff if either methods were used.” They seem to have ignored my statements concerning the requirements for staff expertise (cf., pp. 219 and 221).

In their discussion of approach I, Kalush and Johanson say that “the main problem with the basic method is that it is very wasteful of disk space,” seemingly ignoring the statement that “this off-the-shelf approach wastes space promiscuously” (p. 214). And their assertion that “the use of memo fields means that a separate file is created for every MARC record where the memo field is used” indicates a lack of understanding of the way memo fields work in dBASE. Every time a memo field is used in dBASE, it creates a 512 kilobyte block linked to the main record and stored in a “dbt” file; in short, only one file contains all the data from all records using a specific memo field in the main data base.

I never claimed that method II wasted less disk space than the first, and in fact I implicitly stated: “Thus using the *byte-wasting* data base structure [i.e., approach I] presented above” users can create a dependent data base which controls about 10,000 records (p. 217; emphasis added).

Kalush and Johanson, in their discussion of approach III, also seem to ignore statements concerning the article’s purpose. They charge that the approach “eliminate[s] some of the instances of wasted disk space. . . by making basic setup of the data base far more complicated.” I place a similar caveat upon using approach III: “against the advantage of increased record capacity must be placed the significant amount of programming necessary to set up and maintain the multiple data bases” (p.219). Like Kalush and Johanson, I recommend that a staff member be a dBASE programmer in order to pursue this approach.

Regarding their criticism of method IV, Kalush and Johanson’s statement that “this

method requires that dBASE III open and close each of its files whenever a record is loaded or stored” is simply not true. dBASE can have open ten data base files at the same time, significantly cutting down on the time it takes to load a record following approach IV. Rather than Kalush and Johanson’s unsubstantiated “estimate” that “it would take about 2–3 minutes to load a record to the screen,” my own experience has been that with using nine highly complex data bases, such loading requires only a few seconds. I would expect that Kalush and Johanson would know from their experience with *MicroMARC* that no implementation of the MARC format will fit on the screen at once at any given time. The format is so complex that several screens are required to enter information. Approach IV and most conceivable PC programs which claim to the MARC format do not require that a record be “loaded” into RAM memory at one time; such records would usually exceed the 640k limit of DOS RAM (discounting virtual disk memory) and leave no room for the operating system or the data base management program.

Finally, Kalush and Johanson criticize “the goal of creating large data bases [as] not practical.” At this point in the article, I was discussing the upper limits of using dBASE on a network of shared hard disks and gave as a caveat “sharing disks over a network can dramatically slow down performance” (fn. 12). I agreed in the article with Kalush and Johanson that the “current generation of microcomputers” might not be adequate to the needs of large MARC-based local data bases. In any case, few archivists in the country who intend to control their documents at the collection level will have to face anything near 100,000 records.

In short, it seems that most of the criticisms voiced by Kalush and Johanson could have been clipped from my own article.

RONALD J. ZBORAY
University of California, Berkeley