Volume 3
Number 3

ISSN 0730-9295

Information Technology and Libraries

September 1984

CONTENTS

ITLBDC 3(3) 225–328 (1984); Inf Technol Libr

Information Technology and Libraries

Volume 3, Number 3: September 1984

CONTENTS

ITLBDC 3(3) 225–328 (1984); Inf Technol Libr

Information Technology and Libraries

Information Technology and Libraries is the official publication of the Library and Information Technology Association, a division of the American Library Association, 50 E. Huron St., Chicago, IL 60611; *Executive Director:* Donald P. Hammer. The journal is issued quarterly in March, June, September, and December.

Information Technology and Libraries publishes material related to all aspects of library and information technology. Some specific topics of interest are: Automated Bibliographic Control, AV Techniques, Communications Technology, Cable Systems, Computerized Information Processing, Data Management, Facsimile Applications, File Organization, Legal and Regulatory Matters, Library Networks, Storage and Retrieval Systems, Systems Analysis, and Video Technologies. *ITAL* welcomes unsolicited manuscripts. Submissions should follow the guidelines stated under "Instructions to Authors" on page 103 of the March 1984 issue.

Manuscripts of articles, communications, and news items should be addressed to: William Gray Potter, Editor, *Information Technology and Libraries,* 246 A Library, University of Illinois, 1408 W. Gregory Dr., Urbana, IL 61801. Copies of materials for review should be addressed to: Karin A. Trainer, *ITAL Book Reviews,* Yale University Library, 120 High St., P.O. Box 1603A, Yale Station, New Haven, CT 06520. Advertising arrangements should be made with William Z. Schenck, University of Oregon Library, Eugene, OR 97403.

Information Technology and Libraries is a perquisite of membership in the Library and Information Technology Association. Subscription price, \$12.50, is included in membership dues. Nonmembers may subscribe for \$25 per year. Single copies, \$7.50.

Circulation and Production: American Library Association, 50 E. Huron St., Chicago, IL 60611. Please allow six weeks for change of address.

Abstracted in *Computer & Information Systems, Computing Reviews, Information Science Abstracts, Library & Information Science Abstracts, Referativnyi Zhurnal, Nauchnaya i Tekhnicheskaya Informatsiya, Otdyelnyi Vypusk,* and *Science Abstracts Publications.* Indexed in *Computer Contents, Computer Literature Index, Current Contents, Current Index to Journals in Education, Education, Library Literature, Magazine Index, and NewSearch.* Microfilm copies available to subscribers from University Microfilms, Ann Arbor, Michigan.

Second-class postage paid at Chicago, Illinois, and at additional mailing offices. *Postmaster: Send address changes to* Information Technology and Libraries, *50 E. Huron St., Chicago, IL 60611.*

Packet Radio for Library Automation

Edwin B. Brownrigg, Clifford A. Lynch, and Rebecca L. Pepper

INTRODUCTION

The University of California Division of Library Automation (DLA) has recently demonstrated an experimental communications system that uses a blend of radio and digital packet-switching technology to transmit data between the DLA online Catalog, MELVYL, and its terminals, eliminating the need for data cables. This type of communications system, called *packet radio*, has been in use since the early 1970s, but until now has been largely the province of amateur radio operators and the military.

DLA staff demonstrated their experimental system at the ACRL conference in Seattle, Washington, on April 4 through 6, 1984, with the support of Council on Library Resources grant number CLR-2066. A second demonstration was held at the ARL meeting in Colorado Springs, Colorado, on April 26 and 27, 1984. DLA's demonstration system was built around the Tucson Amateur Packet Radio Terminal Node Controller (TNC) board (discussed below), and 100-milliwatt 49-MHz FM radios adapted for the TNC board. These demonstrations have created interest within the library community in the potential for "wireless" online catalog terminals. In fact, packet radio can be used to create local area networks that include terminals, personal computers, and large mainframes, as well as gateways to remote networks and systems.

This article presents a tutorial on the state of the art in packet radio and an annotated bibliography on packet radio and related technologies. The authors would like to acknowledge the efforts of DLA staff members Bennett Price, David Shaughnessy, and David Wills, who worked to develop the experimental packet radio system. In addition, we would like to thank: the many amateur packet radio operators across the country who have provided much valuable advice; a number of people at Bolt Beranek and Newman, particularly Jim Herman; and the Council on Library Resources, especially C. Lee Jones, for their support.

THE PROBLEM OF WIRING

The advent of online catalogs in libraries has resulted in a problem that could not have been foreseen when most library buildings were built—the need for wiring to transmit data between terminals and the online catalog. This is particularly serious in older libraries where there are insufficient conduits, false ceilings are rare, and one faces the prospect of running cables through marble floors. While this problem has existed since the early days of library automation, the proliferation of terminals for public access systems has made it a severe financial and logistical concern. Looking into the future, as more and more services are distributed electronically, the wiring problem will grow to gargantuan proportions.

Installing data cable wiring can be costly. The experience of the University of California demonstrates that the cost of installing terminals in quantities of eight to ten may range from $8,000 to $12,000, not including staff costs. Even if the wiring for data communications makes up only half of this figure ($4,000 to $6,000), it is evident that the wiring can cost as much as or more than the terminal itself[1].

In addition, it can take months to ar-

Edwin B. Brownrigg is director and Rebecca L. Pepper is a writer, Division of Library Automation, University of California. Clifford A. Lynch is manager of the Computing Resources Group.

range to install the wiring, and, further, new wiring must be installed when the terminal is moved to a new location, making it costly and time-consuming to relocate online catalog terminals.

Using packet radio for transmitting data to and from an online catalog avoids the expense of installing a cable for each terminal, and it also makes the terminals far more portable. Unlike terminals that communicate across a cable, a packet radio terminal can be installed very quickly.

Online public access systems also give rise to a broader problem in academic settings. The online catalog and its follow-on document delivery capabilities are making it possible to move access out of the libraries and into the other buildings on the campus: faculty offices, laboratories, dormitories, and the like. Wiring an entire campus is a slow and enormously costly proposition. A radio-based network offers an alternative to the wired campus that is rapidly deployable, can be installed in increments, and is lower in cost.

RADIO FOR DATA COMMUNICATIONS

Radio transmission of data is not new. Data has been traveling over point-to-point microwave links and other line-of-sight systems since the 1950s. At first, these systems were oriented toward analog traffic such as voice; digital links appeared later. Data can also travel readily on a one-way broadcast channel such as a television channel. In both of these applications, however, only one user can transmit on a given frequency.

In the late 1960s, the concept of packet communications was developed. Initially it was used to share access to networks of leased lines that connected packet switches such as the ARPANET. In packet communications, the data to be transmitted is divided into chunks called packets. The address of the data's destination is added to each packet, and the packet is then routed to its destination over a series of point-to-point lines through store-and-forward packet switches (called Interface Message Processors, or IMPs, in the ARPANET). Packet communications are widely used today in long-haul telecommunications networks. See Brownrigg and Lynch [2] and Shaughnessy and Lynch [3] in the bibliography below for a discussion of long-haul library telecommunications networks.*

Packet switching in networks based on point-to-point lines accomplished two major objectives: it provided a means of sharing or multiplexing the point-to-point lines among many users, and it provided a convenient unit (the packet) on which switches could perform routing. Still, for any line, there were only two points of access—one on each end of the line—that did all of the multiplexing. All sharing was managed by these end points.

Packet radio applies the principle of packet communications to a broadcast channel. A radio channel can be accessed by hundreds of transmitters simultaneously. This is very common in voice radio applications such as citizens band (CB) radio. Until the introduction of packet radio, however, data transmission over radio channels was typically done as though the radio channel were a point-to-point link. (This was not entirely the case; techniques such as subcarriers were sometimes used to subdivide one channel into several, and some experimental spread spectrum systems were also used to share radio channels. See the papers by Scholtz and Price in Cook [24].)

A packet radio consists of a transceiver and a microprocessor that, when connected to a terminal or microcomputer, allow the device to send and receive data, making possible wireless data communications. Figure 1 is a block diagram of a packet radio unit. A receiver will recognize and accept only packets that bear its address, allowing a number of receivers to listen for data over the same channel. Packetized data also allows a number of transmitters to operate over the same channel by permitting the use of communications protocols—the rules by which computers communicate—that will avoid most packet collisions, and will recognize collisions when they occur and retransmit the packet(s). This is discussed in detail below.

A packet radio system operates on one or at the most two radio channels or frequencies. This distinguishes it from cellular radio (a technique that is beginning to be used

*All numbers in brackets refer to items in the bibliography that follows this article.

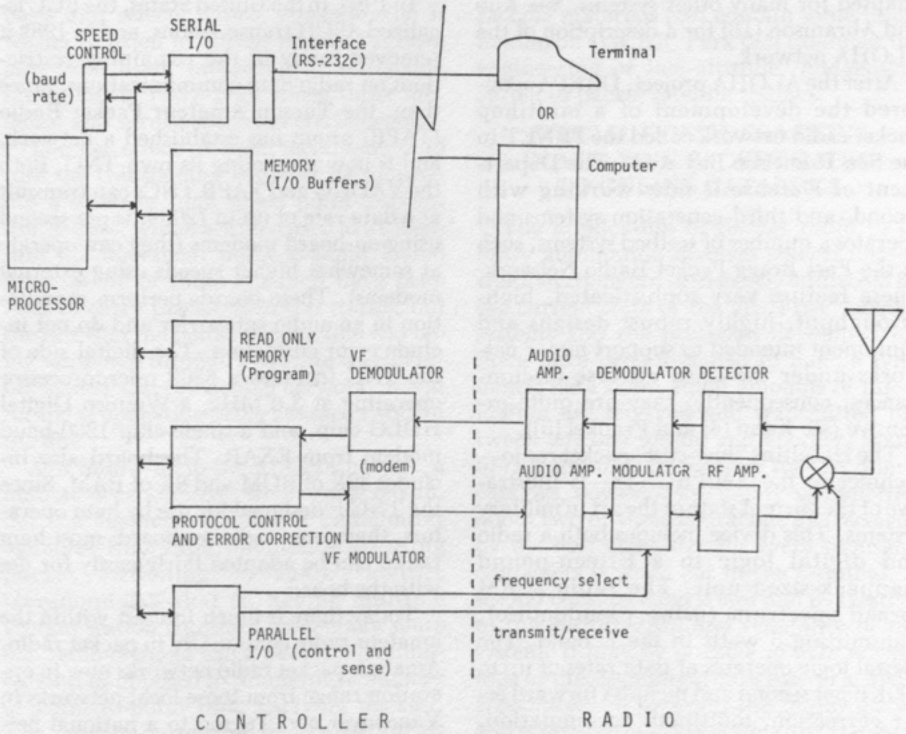


Fig. 1. Packet Radio Unit Block Diagram.

for mobile telephones), in which a large pool of frequencies is maintained, with each caller allocated a frequency for the duration of the call's existence within a cell (see Godin [11]). Packet radio has more in common with some recently announced hybrid systems such as Motorola's portable computer system, which features a handheld computer that can communicate with a remote computer by radio over a single pair of frequencies (see Krebs [12]). However, there are substantial differences in the two systems: Motorola's system does not allow communication between terminals or portable computers except by passing the message through a central site. It is also designed to cover larger geographical areas than those covered by typical packet radio networks.

A BRIEF HISTORY OF PACKET RADIO

Military Activities

The military has been interested in packet radio for more than a decade, primarily as a battlefield communications network that can be deployed rapidly, can quickly adapt to rapidly moving nodes (such as a terminal mounted on a truck or an aircraft), can avoid single points of failure, and is robust in the face of jamming or other interference. The development of packet radio began in the early 1970s with the University of Hawaii's ALOHA packet radio network, a watershed in the development of modern telecommunications protocols. The ARPANET, a large-scale wire- and satellite-based packet communications network developed by Bolt Beranek and Newman for the Defense Advanced Research Projects Agency (DARPA), had already been in operation for several years by then (the ARPANET was established in 1969). The ALOHA network was initially a single-hop system using no repeaters, in which various devices such as terminals, minicomputers, and graphics processors communicated via radio with a central computer. The protocols developed for the ALOHA network were later redefined and adapted for many other systems. See Kuo and Abramson [16] for a description of the ALOHA network.

After the ALOHA project, DARPA sponsored the development of a multihop packet radio network called the PRNET in the San Francisco Bay Area. The Department of Defense is now working with second- and third-generation systems and operates a number of testbed systems, such as the Fort Bragg Packet Radio Network. These feature very sophisticated, high-throughput, highly robust designs and equipment intended to support major networks under the most adverse circumstances; consequently, they are quite expensive (see Kahn [4] and Frankel [5]).

The Hazeltine "low-cost" packet radio—technically the AN/PRC-118—is illustrative of the current state of the art in military systems. This device includes both a radio and digital logic in a fifteen-pound manpack-sized unit. The radio is FM spread spectrum (using pseudonoise), transmitting 5 watts in the L band. The digital logic operates at data rates of up to 400KB per second and includes forward error correction, multipath accumulation, provision for dual data rates for packet headers and data, and HDLC support. The unit includes an 8086 microprocessor, 48K of RAM, and 16K of ROM.

Amateur Packet Radio Activities

At the other end of the packet radio spectrum are amateur radio operators, who have been experimenting for some years with very low cost, low-throughput systems. The first amateur packet radio network in North America was established in 1978 in Vancouver, British Columbia, after the Canadian government, seeking to encourage the use of packet radio, allocated a set of frequencies (221 to 223 MHz and 433 to 434 MHz) for packet and digital transmissions. The Vancouver Amateur Digital Communication Group (VADCG) soon began to produce and sell a packet radio terminal node controller (TNC), a microprocessor combined with memory that allows a terminal or microcomputer to communicate via radio with other similarly equipped devices.

In 1980, in the United States, the FCC legalized ASCII transmissions, and in 1982 it removed many of the remaining restrictions on radio data communications. Since then, the Tucson Amateur Packet Radio (TAPR) group has established a network, and is now marketing its own TNC. Both the VADCG and TAPR TNCs can transmit at a data rate of up to 1200 bits per second using on-board modems (they can operate at somewhat higher speeds using external modems). These boards perform modulation in an audio subcarrier and do not include error correction. The digital side of the TNC includes a 6809 microprocessor operating at 3.6 MHz, a Western Digital HDLC chip, and a single-chip 1200-baud modem from EXAR. The board also includes 32K of ROM and 8K of RAM. Since the TNC is designed for use by ham operators, there is no radio on-board; most ham radios can be adapted fairly easily for use with the board.

Today there is much interest within the amateur radio community in packet radio. Amateur packet radio networks now in operation range from these local networks in Vancouver and Tucson to a national network that uses a system of repeaters. Several conferences on amateur radio computer networking have been held, and the Amateur Radio Research and Development group has issued preliminary protocol standards for packet radio networks.

Rouleau and Hodgson [7], Morrison and Morrison [8], Johnson [9] and the American Radio Relay League [10] provide an overview of the history and current state of the art in amateur packet radio technology.

Packet Satellite Systems

A communications satellite can be considered a radio repeater in geosynchronous orbit, albeit a very expensive and wideband repeater. As such, users sharing access to a satellite channel face the same problems as users attempting to share any radio channel. Historically, this has been handled by frequency division multiplexing—essentially splitting the channel into a number of fixed-size subchannels. Each ground station then transmits over one or more of these fixed-size subchannels on a dedicated basis and selects which subchannel(s) it wishes to listen to. Other techniques have also been used, including time division multiplexing (very high speed burst modems) and spread spectrum transmission.

The net effect of these techniques is to split the transponder capacity by simulating a series of high-speed point-to-point linked lines, which are then used as a substitute for terrestrial point-to-point leased lines. Some of these channels have been used to carry packet switched traffic, but the fact that the point-to-point line was being simulated through a satellite was invisible to the packet switching network, except for the propagation delay characteristics of the line.

As a follow-on to the ALOHA project, a number of experiments were conducted under the auspices of DARPA and similar agencies that experimented with using packet switching to share satellite transponders—that is, having multiple ground station transmitters actually share one radio channel using packet communications, as opposed to simulating a number of point-to-point links carrying packets. The most important of these experiments are the North Atlantic Satellite Network (SATNET), which linked sites in North America and Europe, and the higher-speed Wideband Net in the United States.

It is important to realize that packet satellite systems are not radically different from terrestrial packet radio systems any more than standard communications satellites are different from standard point-to-point radio-based data communications with an intervening repeater. In both cases terrestrial and geosynchronous satellite networks can be seen as the two extremes of a continuous spectrum of system configurations, with a number of intermediate configurations, such as radios based on high-flying aircraft or low-altitude satellites, being equally feasible. The military makes extensive use of airborne repeaters; ham radio operators have experimented with nongeosynchronous OSCAR satellites. See Martin [14] for general information on communications satellites; Jacobs [15] describes packet switching satellite systems such as SATNET. American Radio Relay League materials [10] contain some information on OSCAR. Park [25] examines the engineering of low-altitude, nongeosynchronous packet satellite systems.

SPECIAL CONSIDERATIONS IN APPLYING PACKET RADIO TO ONLINE CATALOGS

The application of packet radio to library automation involves several unique considerations not sufficiently addressed by the current state of the art. First, current systems and the protocols developed for them have been designed to handle symmetric data rates: equal amounts of information are received and transmitted between any one station and another. Many library automation systems such as online catalogs, however, are highly asymmetric: about two or three characters are received by the host computer for every thousand characters it sends out to the terminal. A second consideration is cost and its relation to the performance and reliability of the system. It may be possible to take advantage of this asymmetic data rate to achieve a compromise between the sophisticated and expensive military system and the inexpensive but somewhat unreliable amateur system. For example, one could use less expensive transmitters in the terminals, where data speed is not crucial, and achieve a high data rate in the other direction by placing a high-quality transmitter in the online catalog's base station and sensitive receivers in the terminals. This would provide high performance while keeping costs to a minimum. Current advances in VLSI circuitry for signal processing are also causing radical changes in the cost/performance trade-offs involved in engineering such systems (see Elliot and Kopec [13]).

Finally, there is a special consideration involving repeaters and routing. It is unclear whether the ultimate library packet radio system will have to incorporate repeaters (this is largely a function of the radio technology issues discussed below). In many applications, one is concerned with networks that are limited geographically—a building or a campus. If repeaters are used, it will be necessary to explore different approaches to routing. Some compromise should be possible between the military system, in which the routing must adapt very quickly to fast-moving vehicles, node failures, and other drastic changes in network topology, and the amateur systems, in which the topology is extremely stable, and there is not much concern for automatic selection of alternate routes.

THE TECHNOLOGY OF PACKET RADIO

Protocol Layering and Packet Radio

Throughout the discussion of packet radio systems below, we address two basic questions:

1. What is the best way to share and manage a broadcast channel?
2. How should data be routed in a packet radio network that is large enough to require the use of repeaters?

Each of these questions deals with different layers of the standard model for communications protocol known as the International Organization for Standardization's Reference Model of Open Systems (hereafter called the ISO model). See Zimmerman and Tanenbaum [26] for information on this model. The first question involves the physical and data link layers, and the second involves the network layer. (It should be noted here that telecommunications protocols at DLA generally follow the DARPA protocol model rather than the ISO model (see Shaughnessy and Lynch [3]). These two models are identical at the low levels under discussion here.)

A brief description of these layers of protocol will help to put these two questions into perspective.

The ISO model is divided into seven hierarchical layers of protocol, each of which performs a specific communication function. The physical, data link, and network layers are the three most basic layers in the model. They involve the means by which data is transmitted from one point in a network to another. The four higher levels—the transport, session, presentation, and application layers—all involve how to translate the data once it gets there. They set up rules that allow one computer to comprehend data received from another.

The physical layer of protocol involves decisions as to the electronics and mechanics of data transmission. In terms of packet radio, the physical layer determines the radio frequencies used, the power and propagation characteristics of the radios, and the modulation techniques used to transmit the data.

The data link layer takes care of ensuring that data is received correctly. In packet radio, this is where the data is broken into packets, the address of the packet's destination is added, and schemes are devised to prevent packets from colliding, to correct errors when they do collide, and to request retransmission of packets that are garbled or lost.

The first question above, how best to share and manage a broadcast channel, deals with the physical and data link layers.

The second question, how best to handle routing in a network, is addressed in the network layer of the ISO model. This layer involves the means by which packets travel through the network to get to their destination. In a packet radio network large enough to require repeaters—devices that receive and rebroadcast data—the protocols used in the network layer can become very complex because repeaters greatly increase the number of packets in the air and thus greatly increase the probability of data packets colliding.

One of the features that makes packet radio difficult to analyze is that the distinction between the physical, data link, and network layers can become blurred, and these layers interact in particularly complex ways.

DEFINING A PACKET RADIO PROTOCOL AT THE PHYSICAL LEVEL

Decisions at the physical level of protocol involve choosing the right blend of transmission frequencies, modulation techniques, transmitter power, and antenna configurations for a given situation.

1. FREQUENCIES. Unless one is transmitting at very low power (less than one-tenth of a watt), the broadcast frequencies must be allocated and licensed by the FCC. The higher frequencies require line-of-sight transmission, while the lower fre-

quencies bounce the signal off the ionosphere. Virtually all packet radio systems operate on line-of-sight frequencies. The choice of high versus low frequencies determines the type of licenses that the FCC will require, the restrictions on the type of modulation techniques to be used, the transmitter antenna design, and the transmitter power permitted by the FCC. The propagation characteristics of the frequency used will determine the cost of the transceivers. The frequency allocations will determine system bandwidth (capacity).

2. MODULATION TECHNIQUES. This area encompasses two issues. The first is the method used to encode digital data into an analog signal so that it can be transmitted over a radio channel. This function is generally performed by a modem of some sort. It is important here to choose an encoding scheme that minimizes interference and maximizes bandwidth use. Some possible methods are:

- frequency shift keying (FSK), in which a slight variation in frequency indicates whether a bit is a zero or a one;
- quadrature phase shift keying (QPSK), in which wavelengths at different phases represent different combinations of bits;
- pulse code modulation (PCM), where binary digits are conveyed as the presence or absence of a pulse.

The second issue involves the method used to encode the analog signal into an analog carrier (the broadcast channel). Possible methods include frequency modulation (FM), amplitude modulation (AM), and frequency modulation using a single sideband (FM/SSB).

An alternative that involves both of these issues would be to use spread spectrum techniques, which are very resistant to interference. Because interference tends to occur sporadically, either in short bursts, or centered on a frequency, spread spectrum acts to spread the data as widely as possible in order to minimize the amount of data destroyed by these bursts. Spread spectrum techniques can be applied either just before the digital signal is converted to analog, or as an extra input when the analog signal is encoded into the radio channel.

At the digital-to-analog stage, spread spectrum can be applied by incorporating pseudonoise into the digital signal. Pseudonoise is a stream of bits generated by a random number generator. For every bit of data to be transmitted, the encoding device will intermix a certain number of random bits into the data stream. The device receiving the data would use the same random number algorithm and the same starting point, allowing it to separate the actual data from the random bits. Spreading the data out in this manner reduces the probability that short bursts of interference will destroy data. See Pettit [18] and the articles reprinted in [24] for further information on spread spectrum.

At the analog-to-radio-carrier stage, spread spectrum can be applied by switching very rapidly between a number of different broadcast frequencies (called frequency hopping), with each frequency selected at random. Spreading the data across a broad spectrum of frequencies reduces the chance that interference on a specific frequency will destroy data.

While spread spectrum is ideal for library packet radio because of its great resistance to interference, the cost may be too high, and there are complex problems of synchronization between the stations that must be resolved. FCC licensing may also be a problem with spread spectrum modulation, although the FCC has recently indicated that it may deregulate the use of spread spectrum techniques (see "FCC Frees up Spread Spectrum" [27]).

3. TRANSMITTER POWER will determine the geographic area that the system can cover reliably and the relationship between transmitter power, antenna configuration, cost of the transmitter, and the sensitivity (and hence the cost) of the receivers. In addition to propagation in open atmosphere, one must consider the ability of the radio signal to penetrate into buildings.

4. ANTENNA CONFIGURATIONS. Selection and configuration of antennas will depend upon the application. For indoor applications, long-wire antennas could possibly be run along the ceilings and up elevator shafts for the base station, and simple polarized antennas could be used on

the terminal radios. For very short distances, infrared is an interesting possibility. For interbuilding communications, a vertical nondirectional antenna might be used for the base station, with a vertically polarized yagi antenna of short length for the terminal cluster node station.

Ensuring System Reliability/Robustness

The reliability/robustness of the packet radio system will depend on the insensitivity of the radio channel to all of the various problems found within buildings and urban areas. These problems include interference both from unshielded RF sources and from other radio services, multipathing, scattering, signal fading, and possibly intentional signal interference. Spread spectrum techniques offer a well-developed means of minimizing all of these problems; this is one of the main motivations for their use in military communications.

A variety of error-correction coding techniques can also be used, either as a supplement or as a replacement for spread spectrum techniques. These techniques fall into two broad categories: block coding (traditional error-correcting codes) and convolutional coding. Block codes operate by computing a series of error-correction bits on a fixed-size block of data (typically using polynomials over finite fields). These bits are then appended to the data. When the block is received, the error-correction bits are recomputed and compared to the received error-correction bits; if they do not match, an error has occurred on the communications channel, and various algorithms can be used to identify and correct the error. Block codes can be seen as an extension of the familiar parity bit or check digit.

Convolutional codes operate on streams of bits rather than blocks of bits, and can take advantage of situations where quantization occurs, as is the case in radio reception. With a block code, one operates on a strictly algebraic basis on bits that are quantized as zeros or ones upon reception. Some convolutional coding algorithms, on the other hand, can take advantage of the radio receiver's ability to differentiate between "hard" and "soft" zeros and ones. In frequency shift keying, for example, a receiver could classify input samples as "definitely a zero," "definitely a one," "closer to zero than one," or "closer to one than zero." Convolutional coding algorithms such as maximum likelihood decoding can take advantage of this information. Viterbi and Omura [19] give details on both types of codes.

Defining a Packet Radio Protocol at the Link Level

The usual protocols for managing a shared broadcast channel are either some variant of ALOHA (standard or slotted), or Carrier Sense Multiple Access/Collision Detection (CSMA/CD) (see Tobagi [17]). Note that in networks with a central point such as a base station, there may be two channels, one for base-station-to-terminal communications, and one for terminal-to-base-station communications, or all this traffic can share the same channel.

In standard ALOHA, transmission is done at any time, and the receiver sends an acknowledgment to the sender for each packet it receives. The sender detects problems if, after a given time interval, it has not received an acknowledgment from the receiver. The sender then delays for a random interval and retransmits. In slotted ALOHA, transmissions may only begin at the beginning of a time interval or "slot"; errors are detected in the same manner as in standard ALOHA. (This has implications for packet length size distributions if it is to make effective use of the channel.)

In the CSMA/CD protocol, the transmitter first listens to the channel to see if it is in use and also listens while it is transmitting, to detect collisions. In a network with repeaters (i.e., where every terminal cannot hear every other terminal), attempts to listen for collisions can be dangerous. A source node may be unable to hear a collision occurring at a destination node, or may "hear" a collision locally that the destination node cannot hear. There is also a problem, but a less serious one, in sensing if the channel is busy. There are a variety of proposals to resolve this, such as transmitting a busy tone at higher power than standard data transmission, but they are all fairly complex. Another variant is straight CSMA, where the transmitter tests the channel to see if it is clear, but does not listen while it is transmitting. With CSMA, collisions are detected only when the receiver does not acknowledge a packet.

Simple ALOHA may be sufficient for terminal-to-base-station transmissions, given the low data rates for this channel. There are well-known analyses of all of these protocols in cases where the data rate is symmetric, but the extent to which asymmetric data rates affect these analyses is not entirely clear.

Other hybrid strategies are also possible, including reservation strategies such as Priority-Oriented Demand Assignment (PODA) (see Jacobs [15]) for part of the channel capacity. This would allow activities, such as file transfer, that generate a steady stream of data for a period of time to proceed parallel with, but not interfere with, bursty traffic such as terminal interactions being carried in the part of the channel capacity being managed on a contention basis.

DEFINING A PROTOCOL AT THE NETWORK LEVEL

A packet radio network with no repeaters, in which a node can directly communicate with every other node, is called a *full broadcast network*. In a full broadcast network, the network level protocol is fairly simple, since there is no issue of routing. The simple act of transmission causes the message to be routed to any node that wishes to receive it.

Routing and Repeaters

Routing is only an issue if the network is large enough to require repeaters. Networks that require repeaters for end-to-end communications are called *semibroadcast networks*.

In semibroadcast packet radio networks, routing algorithms will be necessary in order to control the repeating of packets. There are several approaches to routing, including broadcast routing, path finding, and duct routing.

Broadcast routing is very reliable, but it requires a large system bandwidth if the maximum number of hops between any two points in the network is more than two or three. In broadcast routing, the maximum number of hops between any pair of nodes in a network is calculated, and that count is added to the header of each packet as it enters the network. Each time a repeater receives a packet, it will decrement the count by one, and then will repeat the packet only if the count does not now equal zero. While this eliminates a great deal of unnecessary broadcasting of packets, this approach causes each packet to be broadcast throughout the network. There are a variety of techniques that can be used to further reduce the unnecessary repetition of packets. For example, a repeater could remember the last few packets it has transmitted and not retransmit these again should it receive them.

Path finding is a set of techniques whereby a source node identifies a path through a series of repeaters to the destination node by either using broadcast routing tables or by broadcasting "probe" packets until it can find enough repeaters to reach the destination. Once the path is established, this is the most efficient way to route packets, since a packet is not repeated any more than is necessary.

The problems with path finding are that locating the path can be quite expensive and difficult, and if something happens to disrupt the path (either a repeater failure or some change in the radio propagation topology due to mobile nodes or other factors), this break in the path can be difficult to recognize quickly and requires repetition of the path-finding algorithm to discover a new path. Kahn [4] discusses path finding and broadcast routing in some detail. Yemini [20] provides a good discussion of the performance degradation that can be caused by over-repeating of packets.

In duct routing, a path is chosen between the sender and receiver. The path must be wide enough to include several different routes for the packets, to provide redundancy in case one or more of the repeaters in the path fails. While the concept of duct routing is fairly simple, it is difficult to implement because radios don't understand geography. The spatial universe of a radio is much different from the spatial universe of a human being. Schachan and Craighill [6] discuss duct routing.

Another issue in routing within a network is whether to use the virtual circuit or the datagram approach. In the virtual circuit approach, once a route between sender and receiver has been established (by one of several different methods), that route is used for the duration of the transmission. The packets are sent and received in order. Duct routing and path finding, for example, fit comfortably with virtual circuit approaches as long as the topology of the radio propagation pattern does not change often (due to mobile nodes or other factors).

In the datagram approach, each packet is considered an individual entity, and packets may travel in the network independently of the other packets in its data stream. In this approach, packets may be received out of sequence. The datagram approach offers more flexibility and adaptivity, but at a considerable cost in the case of path finding or duct routing algorithms.

Routing in very large semibroadcast networks seems to be an intrinsically difficult problem, and remains an active area for research.

SECURITY IN PACKET RADIO NETWORKS

Because packet radio networks transmit through broadcasting, they are unusually vulnerable to eavesdropping (interception), particularly if spread spectrum techniques are not used. In addition, it is relatively easy to introduce bogus messages into the network. Because of this, security considerations are of particular importance in packet radio networks.

The major technique for ensuring security is encryption. This can be used both to protect privacy and (in conjunction with a packet sequence numbering scheme) to prevent the introduction of bogus messages or the reinsertion of previously captured legitimate messages.

Because of the possible presence of repeaters, encryption should be done end to end within the packet radio network. Note that destination addresses thus need to be sent in the clear, which means that the network is still vulnerable to traffic analysis. This can be corrected as well, but at the cost of considerable additional complexity, and is not pursued further here, since traffic analysis is not a serious concern in the library automation environment today.

Clearly, an encryption function is needed. In classical cryptography, one defines two functions (algorithms), which encode and decode data. Both operate using the same secret key on any given connection. The sender uses the encryption algorithm with the secret key and the data; the receiver uses the same key with the decryption algorithm to decode the data again. For general use, a reasonable algorithm would be the National Bureau of Standards Data Encryption Standard (DES), although any other encryption function of suitable cryptographic strength for the intended application will work equally well. One of the primary advantages of DES is that while it is computationally intensive if implemented in software, VLSI chips are available that perform the algorithm in hardware.

The difficulty in using DES is that for two parties to communicate they must have prior agreement on the secret key to use. It is clearly undesirable to define N^2 secret keys for an N node network and store N-1 keys in each node, one for each possible node that node might wish to talk to. This is a management problem, since all nodes have to be updated each time a new node is added; moreover, it is a security problem, since keys should be changed regularly.

Ideally, any pair of nodes wishing to communicate should be able to obtain a working session key from a network key server (or otherwise agree on a session key) at will in a secure fashion. Public-key cryptosystems provide a means of doing this.

The basic idea behind public-key cryptosystems is this: each node has two keys—a public key (P) that is published or otherwise known throughout the network (say, from a key server), and a private or secret key (S) known only to the node. A single function (algorithm) f with the following properties is then defined.

for every message (M)

$$f(f(M,P),S) = M$$

$$f(f(M,S),P) = M$$

In addition, let (S1,P1) and (S2,P2) be two key pairs. Then:

$$f(f(f(f(M,P1),S2),S1)P2) = M$$
$$f(f(f(f(M,S1),P2),P1),S2) = M$$

In other words, the function needs to commute. Obviously, it should be very difficult to compute the secret key given knowledge of the algorithm and the public key. Finding such functions is difficult; perhaps the best current proposal is the Rivest-Shamir-Adleman (RSA) system based on the difficulty of factoring large numbers.

Public-key cryptosystems have a number of interesting properties. The first is that there is no need for prior exchange of a secret key; the sender needs only to know the public key of the recipient in order to encrypt. The second is that it is possible to digitally sign a message through the following type of technique:

Suppose node 1 with keys (P1,S1) wishes to send a message to node 2 with keys (P2,S2) so that node 2 can be sure it came from node 1. Node 1 can compute a signature by calculating *f*('Signed Node 1',S1) and then encrypting both this signature and the message using *f*(message || signature,P2).

Node 2 can then decrypt the incoming information into message plus signature by computing *f*(incoming,S2). The signature portion can then be verified by computing *f*(encrypted signature,P1) and looking at the result.

This technique can be used to exchange secret keys between any two nodes in order to set up the DES key for a session. Each node can select its own working DES key and tell the node it wishes to communicate with using the public-key system. The advantage of selecting a DES key rather than just using public-key encryption throughout the interchange is that the DES encryption, implemented in hardware, is fast. RSA is a very complex algorithm that is very slow even when implemented in silicon.

Meyer and Matyas [23] provide much more information on encryption.

GEOGRAPHIC COVERAGE IN A PACKET RADIO NETWORK

By using different types of radio technology, it would be possible to construct a hierarchical packet radio internet to cover a campus building by building. This could be done by placing all of the devices in a given building on a single low-power, limited-size packet radio network. This network could include one or more gateways to a higher-speed campuswide packet radio network (presumably with more expensive nodes due to its higher speed). This campuswide packet radio network would in turn have one or more gateways to the long-haul packet switching network.

What are the real size limits on a packet radio network? By installing enough repeaters, any packet radio network can be extended indefinitely. However, economic and engineering considerations suggest that different designs come into play for different sized networks. Within a building, very low power radios can be used; these may not need licensing at all. For a campuswide network, more powerful radios are more appropriate; these would very likely require licensing.

If one assumes that most traffic is local to some region, such as a campus, then it makes sense to treat networks beyond a certain size as an internet of separate networks linked by gateways.

INTEGRATING A PACKET RADIO NETWORK WITH THE BACKBONE NETWORK

Packet radio is primarily a local technology. At the University of California, for example, we are viewing it as a means of providing access to the online catalog on certain buildings on a campus, or perhaps on a campus as a whole. Therefore, it must be tied to a more traditional long-haul backbone network—at DLA, the network linking the campuses across the state to the DLA computer center in Berkeley. The nature of this interconnection is somewhat dependent on the architecture of the long-haul network in question, and this section is oriented toward DLA's long-haul Transmission Control Protocol/Internet Protocol (TCP/IP)–based packet switching network described in Brownrigg and Lynch [2] and Shaughnessy and Lynch [3], although many of the same considerations arise in connecting packet radio networks to any long-haul network.

1. The simplest means of integrating a packet radio network with a long-haul net-

work is to keep the existence of the packet radio network hidden from the long-haul network by simply plugging the packet radio network into the RS-232 interfaces on a terminal access controller (TAC), and having the packet radio network appear to the long-haul network as a group of terminals. While conceptually simple, this approach creates a mass of wiring and extra hardware at the TAC, since data from the terminals will have to be depacketized and fed into the RS-232 interfaces, only to be repacketized by the TAC. Also, it does not allow many of the sophisticated routing features that would be available if it were treated as a local-area network and became a formal part of the internet. Finally, it means that all users of the packet radio network must appear to the long-haul network as terminals, even when they are using computers, preventing computer-to-computer communication.

2. A second approach would still treat the packet radio network as an "invisible" network (e.g., not part of the internet). It involves building a specialized interface into the TAC in order to eliminate the extra cabling and RS-232 interfaces and perhaps allowing the TAC to support terminals that appear to be passing data through an X.25 packet assembler/disassembler (PAD). This can in many ways be viewed as a more elegant version of method 1.

3. The final approach is to treat the packet radio network as an actual network in the internet. This is the most complex but also by far the most flexible approach. It involves two tasks. First, a gateway must be put in place between each packet radio network and the nearest interface message processor (IMP). Building such gateways is not trivial, as they implement a substantial amount of logic and protocol (see Postel [21] and Strazisar [22]).

The second consideration in treating a packet radio network as a true network arises in the end-to-end protocols that must be used. In order to gateway to the internet, the packet radio network must run standard Department of Defense Internet Protocol (IP) above whatever network-level protocol it uses. In addition, the higher-level protocols that are understood on an end-to-end basis throughout the network (normally TCP and TELNET) must be used on top of IP. The implication here is that either a packet radio TAC must be developed (providing TCP and TELNET for packet radio clients), or each packet radio must effectively function as a "host" on the network, implementing TCP/IP itself. It might be possible to take both approaches: a TAC or terminal server for simple terminals, with the option that intelligent hosts (such as personal computers) on the packet radio network could run TCP/IP themselves and communicate directly with the gateway, allowing them to talk to hosts that are not on the local packet radio network. Some argument could be made for incorporating the packet radio TAC into the base station, possibly by modifying a standard TAC.

COMPONENTS OF A PACKET RADIO SYSTEM

Before examining the components needed to build a packet radio system, we need to review some device interfacing issues. There are two general classes of devices that can be attached to a network: terminals and intelligent devices (computers—micro or mainframe). One can build a network intended to support only terminals, or one that accommodates computers on a peer-to-peer basis. DLA's long-haul network is a peer-to-peer network of computers. Unintelligent terminals gain access to a computer only through direct connection. There is a special fake host computer called a TAC that does nothing except mediate for a group of terminals; most terminals are directly connected to TACs.

This leads to the following components:

1. An interface to connect a host computer to a packet radio network.

2. An interface to connect a terminal or cluster of terminals to a TAC or terminal server through the packet radio network. Note that for closely grouped terminals it may be worthwhile to develop a cluster controller, both for economic and for engineering reasons. This would be a terminal adapter that incorporated several sets of terminal interface logic plus some multi-

plexing logic. Economically, this would avoid having to duplicate the radio transceiver and logic to manage the contention radio channel, reducing the per-terminal cost. From an engineering point of view, it avoids problems of receiver overload, desensitization, and cross talk, which could arise if several independent radios were placed right next to each other.

3. A TAC or terminal server that can mediate on behalf of terminals by supplying the TCP and TELNET protocols needed for end-to-end connections across the internet. This TAC itself might use the interface described in item 2 to connect its terminals and the interface described in item 1 to connect the TAC to the packet radio network.

4. Perhaps repeater switches for the packet radio network.

5. Gateways to link the packet radio network to long-haul networks.

CONCLUSION: PACKET RADIO IN PERSPECTIVE

Packet radio offers some unique capabilities and fills a gap in the current options for local area networking. At the same time, it is important to realize that packet radio is only one alternative for constructing local networks, and, like all other technologies currently in use (baseband, broadband, twisted pair, or fiber optic), it has its strengths and weaknesses. It is strongest in environments that are physically difficult to cable and for applications that require easy reconfigurability and moderate data rates (say under 250KB/second). Specific applications that require very high data rates (say 10MB/second) may be better served by cable- or fiber-optic-based networks. A large library with very complex computer and telecommunications systems may actually need multiple networks: a packet radio system for slower-speed devices and a high-speed network linking a few major computers, with appropriate gateways between the two local networks.

Beyond the engineering issues lie broader questions of overall system architecture and management and of the most effective ways to integrate packet radio networks into the existing library automation environment. All local network technologies merely provide data paths; distributed systems and application-level protocols must still be developed to utilize these data paths to provide services.

With the growth of online catalogs, electronic publishing, and other computer-based public access services, the library of the future will be densely populated with terminals, mainframes, and microcomputers. The cost—both in actual dollars and in administrative complexity—of providing interconnecting wiring for all of these devices promises to be staggering, particularly when wiring is installed in an incremental and evolutionary fashion in buildings that were never designed to house such facilities. Demand for these public access services, at least on university campuses, will spread beyond the immediate library building, leading to an even larger and less tractable wiring problem.

Packet radio technology offers the network designer a new way out of this dilemma. It is not the only way, but in many situations it may be not only economical but also relatively quick to deploy. Given the dynamic state of library automation today and the growing demand for public access, *now*, we believe that packet radio will quickly find its place in the network designer's tool kit.

PACKET RADIO AND RELATED TECHNOLOGIES: AN ANNOTATED BIBLIOGRAPHY

1. Wingfield, M., and R. Edmiston. *Report No. 4810: Telecommunications Study for the University of California Division of Library Automation*. Cambridge, Mass.: Bolt Beranek and Newman, Inc., 1981.

This report, prepared by BBN under contract to the University of California, gives an excellent overview of the options and costs of deploying a large terminal network.

2. Brownrigg, Edwin, and Clifford Lynch. "Development of a Packet-Switching Net-

work for Library Automation." In *Proceedings of the National Online Meeting, New York, April 1983*, p.67–74. Medford, N.J.: Learned Information, Inc., 1983.

3. Shaughnessy, David L., and Clifford A. Lynch. "Telecommunications for an Online Catalog." *Information Technology and Libraries* 2:73–86 (Mar. 1983).

These two articles provide an overview of packet switching telecommunications for libraries. The second continues a good list of references on protocols, internetworking, and routing.

4. Kahn, Robert, and others. "Advances in Packet Radio Technology." *Proceedings of the IEEE* 66:1468–96 (Nov.. 1978).

This is the definitive article on sophisticated packet radio systems (up to 1978), particularly from a military point of view. This very readable article contains excellent discussions of relevant radio technology, an introduction to spread spectrum radio, and a good account of routing and system integration issues. It is probably the best starting place for anyone seriously interested in packet radio.

5. Frankel, Michael. "Packet Radios Provide Link for Distributed Survivable Command Control Communications in Post-Attack Scenarios." *Microwave Systems News* 13:80–108 (June 1983).

This article updates the 1978 Kahn article regarding developments of military packet radio. While largely concerned with broader issues of military command control communications, it contains an interesting discussion of the new "low-cost" army packet radio developed by Hazeltine (note that low cost here is about $12,000). Further details on the low-cost army packet radio can be found in "Inside the News," *Electronics*, 56, no.25:106 (Dec. 15, 1983).

6. Schacham, N., and E. J. Craighill. "Dynamic Routing for Real-time Data Transport in Packet Radio Networks." In *Proceedings, INFOCOM 82, March 30–April 1, 1982, Las Vegas, NV*, p.152–60. New York: IEEE, 1982.

This article updates Kahn's 1978 article in the area of routing algorithms for packet radio networks that include repeaters. It examines hybrid distributed routing schemes such as duct routing.

7. Rouleau, Robert, and Ian Hodgson. *Packet Radio*. Blue Ridge Summit, Pa.: TAB Books, 1981.

This book is a nontechnical introduction to packet radio written for the radio amateur. It presumes a mild familiarity with ham radio but not with computers. Included are descriptions of several amateur packet radio networks and much how-to information. As with all of the ham literature, the focus here is on radio/computer interfacing and low-level protocols rather than the broader questions of system integration and routing.

8. Morrison, Margaret, and Dan Morrison. "Amateur Packet Radio." *Ham Radio* 16 (July 1983), p.14–18; (Aug. 1983), p.18–29 (two-part series).

9. Johnson, Lyle. "Join the Packet-Radio Revolution." *73 Magazine* (Sept. 1983), p.19–24; (October 1983), p.20–31 (two-part series).

Two tutorial articles on recent developments in amateur packet radio, written for the ham radio operator.

10. American Radio Relay League. *Proceedings, Second ARRL Amateur Radio Computer Network Conference, March 19, 1983, San Francisco*. Newington, Conn.: ARRL, 1983.

A very valuable mix of tutorial papers, research reports, and proposals for future developments in the ham packet radio community. Additional references on ham activities include the *Proceedings of the First ARRL Amateur Radio Computer Networking Conference, October 16 through 17, 1981, Gaithersburg, Maryland*, volumes 1 and 2 (although some of this material is now dated by recent developments) and the bulletins from the Pacific Packet Radio Society. In addition, the American Radio Research and Development Corporation (P.O. Drawer 6148, McLean, VA 22106-6148) publishes a newsletter that includes short articles and reviews of ham activities in the digital radio arena generally, with good coverage of packet radio hardware and software and spread spectrum research.

American Radio Relay League. *The Radio Amateur's Handbook*. Newington, Conn.: ARRL, published annually.

A comprehensive sourcebook for ham radio technology, published annually by the ARRL. An excellent reference for terminology in ham radio articles, antenna design, propagation, and other matters.

11. Godin, Roger J. "The Cellular Telephone Goes On Line." *Electronics* 56, no.19:121–29 (Sept. 22, 1983).

A good, readable explanation of cellular radio. More details can be found in Stan Prentiss, *Introducing Cellular Communications: The New Mobile Telephone Systems* (Blue Ridge Summit, Pa.: TAB Books, 1984).

12. Krebs, Jay. "Portable Computer and Host Talk over Radio-Frequency Link." *Electronics* 56, no.17:142–45 (Aug. 25, 1983).

This is a description of the cellular radio system developed by Motorola, Inc., and being deployed by IBM to communicate with its field engineering force. This system is an interesting hybrid that combines ideas from cellular and packet radio.

13. Elliot, Dane, and Stan Kopec. "One Chip Carries out Ethernet Protocol." *Electronic Design* 30, no.20:121–28 (Sept. 30, 1982).

This article provides a summary of the impact of current VLSI technology on the development of special-purpose chips for networking support and a survey of some of the currently available chips. The March 1984 issue of *Digital Design* has a useful and more up-to-date table of VLSI LAN interfaces on page 72. For a very good discussion of the impact that VLSI chips are having on signal processing and of the kinds of things that can now be done economically with digital signal processing chips, see the Special Issue on Signal Processing Technology, *Electronic Design* 32:10. See also Carver Mead and Lynn Conway, *Introduction to VLSI Systems* (New York: Addison Wesley, 1980). This book discusses advances in VLSI design methodology and the growing ability to translate complex algorithms into silicon.

14. Martin, James. *Communications Satellite Systems*. Englewood Cliffs, N.J.: Prentice-Hall, 1978.

While oriented primarily toward long-haul satellite systems, this book provides a good, not overly technical account of some of the major techniques for managing broadcast radio channels. Martin gives short shrift to packet satellite techniques, however, which are fully covered in the article "General Purpose Packet Satellite Systems" (below).

15. Jacobs, Irwin, and others. "General Purpose Packet Satellite Networks," *Proceedings of the IEEE* 66:1448–67 (Nov. 1978).

This is a key paper on developments (to 1978) in packet satellite technology and protocols for the management of broadcast satellite channels. Some of the considerations also apply to terrestrial packet radio networks.

16. Kuo, Franklin F., and Norman Abramson. "Some Advances in Radio Communications for Computers." Report prepared for Air Force Office of Scientific Research, 1973 (NTIS Document #AD 761543).

This technical report describes the ALOHA system at the University of Hawaii, one of the first packet radio systems.

17. Tobagi, F. A., "Multiaccess Link Control." In *Computer Network Architectures and Protocols*. 2d ed., p.145–89. Ed. by Paul E. Green, Jr. New York: Plenum Press, 1982.

This is a survey and tutorial article dealing with protocols for sharing broadcast channels. All of the popular protocols, such as the various forms of ALOHA, CSMA, etc., are described here. It provides a useful summary of what is known about the performance of various protocols and a good list of references.

18. Pettit, Ray H. *ECM and ECCM Techniques for Digital Communications Systems*. Belmont, Calif.: Lifetime Learning Publications, 1982.

A concise and readable introduction to spread spectrum techniques, modulation, cryptography, and error correction using both convolutional and block codes.

19. Viterbi, Andrew J., and Jim K. Omura. *Principles of Digital Communication and Coding*. New York: McGraw-Hill, 1979.

A thorough and detailed reference on error correction on digital communications channels through both convolutional coding and more traditional block error-correcting codes.

20. Yemini, Yechiam. "On Channel Sharing in Discrete Time Multiaccess Broadcast Communications." Ph.D. diss., University of California, Los Angeles, 1980. Distributed by the UCLA Computer Science Department as Report #CSD 830606 and available from NTIS as document #ADA130783.

This Ph.D. dissertation provides a very detailed analysis of the performance of various access algorithms for packet radio networks that include repeaters and a good survey of prior work in the area. It is somewhat technical.

21. Postel, Jonathan B. "Internetwork Protocol Approaches." *IEEE Transactions on Communications* 28:604–11 (Apr. 1980).

22. Strazisar, Virginia. *How to Build a Gateway*. IEN 109. Cambridge, Mass.: Bolt Beranek and Newman, Inc.: 1979.

These two references discuss both the philosophy behind internet gateways and the algorithms necessary to build one. The article by Sheltzer and others, "Connecting Different Types of Networks with Gateways," *Data Communications*, 12, no.8:111–122 (Aug. 1982), provides a good description of how gateways are actually used in the DARPA internet.

23. Meyer, Carl H., and Stephen Matyas. *Cryptography: A New Dimension in Computer Security: A Guide for the Design and Implementation of Secure Systems*. New York: Wiley, 1982.

A good discussion not only of cryptographic algorithms such as DES (the NBS Data Encryption Standard), but also of key management and distribution strategies, techniques, and problems. See also Whitfield Diffie, "Cryptographic Technology: 15 Year Forecast," *SIGACT News* 14, no.4:38–57 (Fall/Winter 1982). This is a good overview of developments in cryptography from more of a survey point of view than the Meyer and Matyas work, which is a textbook.

24. Cook, Charles E. and others, eds. *Spread Spectrum Communications.* New York: IEEE, 1983.

This is a very useful compendium of reprints of key papers on spread spectrum communications.

25. Park, Ke Y. "Global Packet Radios Using Low Altitude Satellites." In *IEEE 1980 National Telecommunications Conference, Houston, Nov. 30–Dec. 4, 1980,* p.70.5.1–70.5.5 New York: IEEE, 1980.

26. Zimmerman, H. "OSI Reference Model—the ISO Model of Architecture for Open Systems Interconnection." *IEEE Transactions on Communication* COM-28:425–32 (Apr. 1980).

See also Tanenbaum, Andrew. *Computer Networks.* Englewood Cliffs, N.J.: Prentice-Hall, 1981. This is a good general textbook on network protocols.

27. "FCC Frees Up Spread Spectrum." *Electronics* 57, no.10:50 (May 17, 1984). ■■

Decision Support System in the Management of Resource-Sharing Networks

Yogendra P. Dubey

In recent years, decision support systems (DSS) have emerged as a tool for decision making. The major forces contributing to the development of DSS have been an understanding of the nature of decision making itself and the sophistication of computer technology (hardware and software), making possible the integration of the computer's information processing capability with human decision-making processes. Two categories of DSS have been identified—data oriented and model oriented. Model-oriented DSS uses simulation and suggestion techniques. Simulation is not unknown to the field of library and information science. Its application, however, has been limited to the study of specific library operations such as circulation and not to the development of DSS for library managers. Recently, the School of Library and Information Science, University of Pittsburgh, has developed a DSS using simulation modeling techniques to aid library managers in their decision-making activity, operating under the complicated environment of resource-sharing networks. The system will be helpful to library managers in long-range as well as short-range planning and policy formulation.

The decision support system is a relatively recent development in information systems, representing a turning point in the utilization of the computer as a decision-making tool. Computer-based decision support systems, although slow to evolve, represent a radical departure from the traditional business applications such as electronic data processing and management information systems. A recurring theme in the literature of management science is that management information systems have been disappointing insofar as they commonly have failed to meet the expectations of managers.[1] The purpose of creating management information systems has been to obtain increases in productivity and effectiveness in organizations. However, they have been unable to achieve this purpose, failing to provide the conceptual power that could mitigate the cognitive constraints of human decision makers in exploring problems and finding solutions. Indeed there is evidence that the information that managers use in critical decisions does not emanate from formal management information systems.[2] Wagner argues that for management information systems, operations research, and other analytic disciplines, the need is to bridge the gap between their specialized world and that of the managers.[3]

In the 1970s the decision support system (DSS) emerged as a new, practical approach for applying computers and information to problems facing management.

Yogendra P. Dubey is a doctoral candidate at the School of Library and Information Science, University of Pittsburgh.

Although the term *decision support system* bears many connotations, the principal tenets characterized by Keen are

1. The emphasis is on decisions in which there is sufficient structure for the use of computer and analytic aids to be of value, but where the manager's judgment is essential.
2. The payoff is in extending the range and capability of the managers' decision process to help them improve their effectiveness.
3. The relevance for managers is the creation of a supportive tool, under their own control, which does not attempt to automate the decision process, predefine objectives, or impose solutions.
4. The system is able to respond quickly to the changing needs of the decision makers.[4]

The basic feature of DSS is to provide support for decision-making activities. The system affords an interface between the decision maker and the computer aimed at simplifying the steps from problem formulation to problem solution. The DSS assists the individual in arriving at an effective decision in a way that contains elements of both subjectivity and objectivity, elements important in the resolution of most problems. The capacity to combine the individual's judgment about a problem (subjectivity) with the computer's output (objectivity) provides the ability to look inside a problem to see what makes it tick.[5]

INFORMATION AND DECISIONS

All managerial activity in an organization revolves around decision making.[6] Forrester argues that management is the process of converting information into action.[7] The conversion process is called decision making. If management is the process of converting information into action, then it is clear that management's success depends primarily on what information is chosen and how the conversion is executed.

As noted by Yovits et al., information is data of value in decision making and information gives rise to observable effects through decision making.[8] Because of task uncertainty, organizations must process information in order to accomplish work. Uncertainty is the relative difference between the amount of information required to make a decision and the amount of information possessed by the organization. According to Galbraith, the greater the task uncertainty, the greater the amount of information that must be processed by the decision makers in order to achieve a given level of performance.[9]

Two consequences stem from the concept of uncertainty. First, uncertainty cannot be conveniently handled by refinement of management techniques, for instance operations research or management information systems. Second, the amount of information needed in making a decision is related to the degree of uncertainty in the environment.[10] It is clear that there is need for an understanding of information processing as a means to reduce uncertainty.

INFORMATION PROCESSING AND DECISION MAKING IN ORGANIZATIONS

An *organization* has been defined as a set of human beings, procedures, and means of production—such as capital, raw material, and data—trying to achieve goals in some rational manner. To reach these goals, decisions must be made. Decision making in organizations is regarded as a process of problem solving. As human beings operate as decision makers, their host organizations are confronted with the cognitive limitations of human beings.[11]

This view implies some understanding of human behavior in a problem-solving situation. A number of models have been proposed and developed in the past two decades that give an insight into human problem-solving behavior. Newell and Simon's model of the human information processing system (IPS) is of particular significance.[12] Their findings indicate that human IPS involves

1. Processing of symbols that are symbolic inasmuch as they have a sensory referent or are a combination of other symbols.
2. Utilizing two levels of internal memory—long-term and short-term memory. The long-term memory has unlimited capacity but slow access. Short-term memory has very small capacity (five to seven symbols), and

memorization in short-term memory needs rehearsal.

3. Serial processing of symbols.
4. Elementary processing times on the order of fifty milliseconds.
5. The existence of a problem space within the limits set by the task environment. The task environment exists in the real world. The problem space is an abstraction of the real world conceived by the IPS. It is the representation of a task environment.

Human beings as information processors exhibit limitations due to slow access to data, sequential processing, the small capacity of short-term memory, and the reduction and simplification of the real world as abstracted in the problem space. Due to these cognitive limitations, human beings cannot make rational decisions which assume that; (1) all possible alternatives are known; (2) all outcomes are known and; (3) preferences for every outcome can be ordered. Inability to make rational decisions by human beings due to cognitive limitations led Newell and Simon to the concept of "bounded rationality."

Because of bounded rationality, most decision makers resort to problem-solving strategies leading to "satisficing," which means that the goal of any decision maker is to get a good enough answer, not necessarily the best possible one. Problem-solving strategies for "satisficing" are based on heuristic rules of thumb that give solutions that are good enough most of the time. Keen and Morton suggest that one can usually improve one's hueristics, although at the often unacceptable cost of increased effort. A DSS may provide this improvement at a lower cost in terms of cognitive strain.[13]

TAXONOMY OF DECISIONS AND CONCEPTUAL FRAMEWORK FOR DSS

Many authors have analyzed the characteristics of decisions and have identified and proposed classification and typologies that help in understanding the conceptual framework on which DSS has developed. The concept involving DSS was first articulated by Morton.[14] Later Gorry and Morton developed a model of decision making in organizations that served the conceptual framework for the subsequent development of DSS.[15] They developed their model by synthesizing Simon's decision-making typology with Anthony's levels of management activity.[16,17] Their model is a two-dimensional taxonomy of decisions showing each level and type of decision requiring a specific type of information system. The types of decisions are (1) structured, (2) semistructured, and (3) unstructured. The levels of decisions include (1) operational control, (2) management control, and (3) strategic planning. It is the semistructured and unstructured decisions of the management control and strategic planning levels that are of greatest concern to decision makers.

FRAMEWORK FOR INFORMATION SYSTEM IN THE LIBRARY RESOURCE SHARING NETWORK

The matrix given in table 1, adapted from Keen and Morton, with examples from the library resource sharing network environment, shows the levels and types of decisions and the support needed for the decision making.

STRUCTURED, SEMISTRUCTURED, AND UNSTRUCTURED DECISIONS

Structured decisions allow the identification of all the elements in the decision process, permit getting one's hands on all important parameters surrounding the decision process, and are easily quantifiable for determining a rational solution. The approach to task solutions may be simple and straightforward. Because of the routine and repetitive nature, the tasks can be automated and delegated to clerical staff for execution.

Semistructured decisions are those that are hard to standardize, perhaps because of the size of the problem or the computational complexity and precision needed to solve them. The models of management science or data alone are inadequate because the solution should involve some judgment and some subjective analysis.

Unstructured decisions are those that are either not capable of being structured or that have not yet been examined. Problems

Table 1.

Types of Decision	Operational Control	Management Activity Management Control	Strategic Control	Support Needed
Structured	Placing purchase order, cataloging, circulation, processing of ILL	Cost analysis, budget estimates	Location of branch libraries, terminals	Clerical MIS
Semistructured	Acquisition decisions, estimating conversion (catalog) alternatives and costs, search strategy	Forecasting and service promotions, estimating network capabilities of satisfying demands	Centralization/ decentralization, decisions reg. sharing of resources among network members/ network operations	DSS
Unstructured	Determining future need of library users	Purchase vs. resource sharing, hiring and motivating of staff	Planning/policy/ R & D	Human intuition

Adapted from Peter G. W. Keen and Michael S. Scott Morton, *Decision Support Systems: An Organizational Perspective* (Reading, Mass.: Addison-Wesley, 1978).

that contain more variables than human beings can comprehend or variables that are subject to influences one cannot control or predict generally fall in the category of unstructured decisions.

Recognizing the types of decisions has had a major influence on the way computers are used in organizations. With structured decisions, computers are used to process data and produce reports. On the other hand, with semistructured and unstructured decisions, computers are used to increase the decision maker's ability to reach a "satisficing" solution by integrating computer output with the subjective judgments of decision makers. The integration of the computer with the human decision-making process will be discussed in the next section.

COUPLING OF COMPUTER CAPABILITIES AND HUMAN THOUGHT PROCESS

Simon's research on the computer's information processing capabilities has had bearing on the practicality of computerized systems for assisting human decision makers, as the computer is the only means currently available with the potential for handling and replicating human thought processes.[18] Mintzberg found that managers collect and piece together various scraps of information until patterns begin to form a mental model that describes various aspects of a problem.[19] Due to advancement in hardware and software and increased memory size and speed, computers can be programmed in ways corresponding loosely to the faculties of memory and reasoning. For example, certain database management systems, by providing the ability to retrieve information in such a way that meaningful patterns and correlations may be discerned, augment memory. Simulation-model-based systems can be seen as extensions of an executive's reasoning powers. Models of the business environment help an executive envision possibilities for the future, foresee consequences, and identify and select solutions.

A number of computer-based modeling languages have been developed. Among such languages, the Interactive Financial Planning System (IFPS) is important and can be used on an ongoing basis by managers and their immediate staff in direct support of their management activities. The IFPS enables non-data-processing people to deal with problems in an interactive exploratory manner. The model assists managers in planning, resource allocation, overall accountability, long-range planning, and other goal-seeking decisions. The model uses equations and logic (what-if analysis) to express the interaction of the various elements and assumptions of a given activity. One particularly noteworthy aspect of the financial modeling language is the use of Monte Carlo risk analysis. This permits decision makers to express intuitively patterns of uncertainty and immediately gain insight into different alternatives.[20]

SIMULATION-MODEL-BASED DSS IN THE MANAGEMENT OF LIBRARY RESOURCE SHARING NETWORKS

Libraries are witnessing dramatic changes that threaten the status quo. The rising costs of materials and services, scarcity of space and shrinking budgets, growing demands and expectations, and use of technologies are the major changes in libraries and in their organization and management.

In order to mitigate the problems caused by these forces of change, libraries have intertwined themselves in complicated resource-sharing networks. The complexities of networks are due to

1. types of networks (i.e., star, hierarchical, distributed);
2. types of resources (i.e., monographs, serials, nonbook material);
3. operations performed (i.e., cataloging, ILL, circulation, etc.);
4. category of networks and financial type (i.e., public versus private, local versus regional, general versus special, profit versus nonprofit).

The reality of libraries operating under the complex web of network environments has reinforced the necessity of pursuing a more realistic approach to policy management, so as to maximize the benefits of joining networks and to achieve an economic break-even point. In consequence, library managers are faced with measuring the performance of networks and individual libraries because they want to know

1. probability of satisfying a demand;
2. average waiting time to satisfy a demand;
3. average cost to satisfy a demand;
4. full cost of network membership versus independence.[21]

The fundamental problem is that library managers must evaluate alternatives before initiating policies. Among those that have expressed the necessity of such an evaluative study, Rouse and Rouse have stated that "while the benefits of resource sharing seem intuitively logical, library managers need to identify the criteria which can be used to measure the costs and benefits offered by the library networks."[22] How is this to be done? Broadbent[23] raises the question and Williams[24] delineates the factors associated with determining the criteria of measuring the costs and benefits of library networks. However, cost-benefit study would require much recursive computation, which most library managers would tend to avoid due to their lack of statistical training and inclination.[25] Moreover, because of the inherent stochastic processes and the complex interactions of variables, problems of resource-sharing networks would defy standard mathematical or operations research techniques for their solution. Library managers require a tool for decision making that makes it possible to constrain a model of the problem, permitting experimentation, and to evaluate various strategies for the operation of the library resource-sharing networks. The answer, however, is to be found in Kent's proposal for the use of computer modeling and simulation to predict economic break-even points for library functions performed in a network environment.[26] Williams advocates the application of simulation technique to decision making and evaluation in the area of resource sharing.[27] Simulation as described by Shannon is the process of designing a computerized model of a system (or process) and conducting experiments with this model for the purpose either of understanding the behavior of the system or of evaluating various strategies for the operation of the system.[28] The simulation has evolved as a natural extension of operations research and the maturation of the high-speed digital computer. Computer simulation facilitates experimentation without going to the real world, so that a more or less random search for alternatives guided by a few heuristics becomes feasible.

The recent literature of management science shows that computer simulation is increasingly being used as a tool for decision making in the area of business. In his survey, Alter has identified two categories of DSS: data oriented and model oriented.[29] Model-oriented DSS uses simulation and suggestion techniques. DSS using simulation modeling techniques permits evaluation of alternative courses of action based on facts and assumptions with a computerized mathematical model in order to represent actual decision making under conditions of uncertainty.

There is evidence in the literature of library and information science that the potential of DSS as a decision tool in the complex and changing library environment is increasingly being recognized. Heindel and Napier emphasized the role of DSS in decision making in library management.[30] Chorba and Bommer proposed an approach of designing a data-oriented DSS for decision making in the management of academic libraries.[31]

A survey of the literature revealed a number of applications of simulation in the study of library management problems such as forecasting of manpower needs at the circulation desk[32] and computer simulation of a circulation system.[33] The model proposed by Rouse and Rouse is a significant contribution for studying the performance of networks.[34] However, none is related to developing a DSS. Only recently has an attempt been made in this direction.

RESOURCE-SHARING SIMULATION SYSTEM

The School of Library and Information Science, University of Pittsburgh, has developed a resource-sharing network simulation system as a follow-up to the University of Pittsburgh study.[35] This is a decision support system using simulation techniques. The system is designed to reflect and mimic the conceptual model of the resource-sharing network developed for that purpose. The simulation model basically consists of a set of data structures that represent in the computer the structure of the conceptual model, which consists of a number of modules that manipulate the data structure to produce the behavior of the conceptual model.

The conceptual model developed for the system views a network as a collection of nodes joined together for the explicit purpose of sharing resources. Each node may have characteristics that differ significantly from other nodes, but in general it is assumed that there are some underlying commonalities across the nodes of the network.

The philosophy of this network simulation is that the characteristics, policies, and relationship of the nodes define the network nodes, or their selected administrative body, but it is still necessary to describe each node individually. The basic theoretical framework of the model is that a network consists of a set of nodes and the nodes are connected via communication facilities. Each node in the network has a set of demands placed upon it and a set of resources needed for the satisfaction of the demands.

Demands may be explicitly generated by patrons, such as the demand to charge out a particular type of material, or implicitly generated by a library policy, such as a book selection policy. These demands create the impetus for transactions to be processed by the resources available in the library node or in other network nodes. Demands must be stated in terms of transactions. A transaction is a unit of service that must be provided to satisfy the demand.

A single transaction must be logically independent from all other transactions in the same category. That is, if one of several transactions is not processed, it should have no direct effect on those that do get processed. There must be a set of demand categories defined and described for each node in the network. In addition, a named demand category must have the same meaning in all nodes of the network. Not all nodes in the network need to have the same set of demand categories.

Each demand requires that one or more work functions (i.e., activities necessary to fulfill a demand) must be performed to provide the service identified in the demand. The same demand at different nodes may be accomplished using different work functions. However, the same work function name must be defined across the network in the same way.

Each work function consumes one or more resources either from the set of resources at the node or at other nodes in the network. A named work function may consume different resources at the node or other nodes in the network. A named work function may consume different resources at different nodes even though it may have the same name and meaning throughout the network.

Resources are tangible entities such as personnel, terminals, shelf space, disk space, line printer, memory, CPU, etc. At each node there is only one set of resources that the work functions have to share. These resources have a finite capacity (expressed in terms of units). Once the capacity of certain resources are exceeded, no more demands requesting these resources can be satisfied. A named resource must have the same meaning throughout the network, otherwise sharing of resources would be meaningless.

Associated with each demand and each resource are a set of policies. These represent logical and quantitative decisions that affect the model flow. In the case of demand, for example, there is a service policy that dictates whether or not assessing the network is permitted. In the case of a resource, for example, there is a policy which indicates whether or not utilization of the resource units is permitted by other nodes in the network.

An important point about the system is that it does not put any constraints on the user or on the naming of a certain level or on the structure of the network. Nodes, demands, work functions, and resources are defined by the user and are meaningful to that particular user.

The system permits simulation by time periods, which are user defined. A time period could be a second, a minute, an hour, a week, or even a year.

The following is a brief description of the simulation model with a schematic representation of the model (figure 1) as given in an unpublished report.[36] Currently, the simulation consists of the following modules:

1. EDIT: Date Entry
2. MODEL: The Simulator
3. ECOMOD: The Economic Model
4. The Statistical Analysis Model
5. Report Generator.

EDIT permits initialization of the data request to run the network simulator. It has

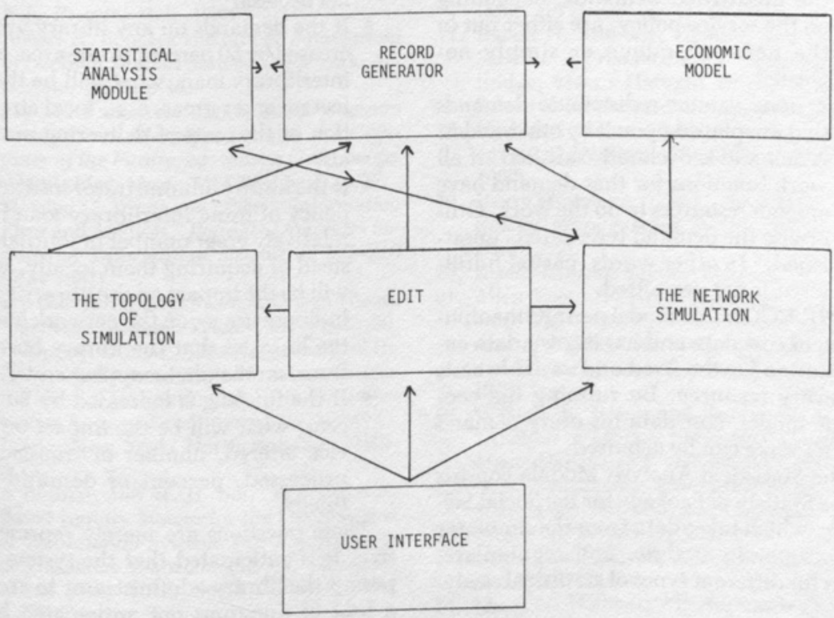


Fig. 1. Schematic Representation of the Resource-Sharing Network Simulation Model.

been designed to run in an interactive mode. It consists of a number of modules, each of which is assigned to handle one level in the conceptual model. For example, one module handles the node, another a demand, and so on. Every module is equipped with the basic editing functions such as adding, deleting, updating, etc.

MODEL is the simulator that permits the user to mimic the behavior of a conceptual model. It examines each node and tries to satisfy the demand by consuming the resources at the node first. If any resource is not available, either due to consumption of all of its units or due to its absence from the node, the simulator tries to satisfy the unsatisfied demands, using resources in the preferred node list. This can only happen if the demand service policy permits network access and the resource policies for the nodes to be accessed allows external nodes to utilize their units.

The design of the simulator model is based on the following general approach and philosophy:

- The computation, i.e., the attempt by the simulator to satisfy the demands, is done in two passes. In pass one, an attempt is made to satisfy the demand using the resources at the node, and all the unsatisfied demands, depending on the service policy, are either put in the network queue or simply neglected.
- A node cannot redistribute demands that are placed upon it by other nodes.
- A demand is declared "satisfied" if all work functions for that demand have enough resources to do the work. Otherwise the demand is declared "unsatisfied." In other words, partial fulfillment is not permitted.

THE ECOMOD model permits manipulation of cost data and has its own data entry routine for the fixed and variable costs for every resource. By running the economic model, cost data for every demand and resource can be acquired.

The Statistical Analysis Module consists of the Statistical Package for the Social Sciences, which takes data from the simulator and economic modules and manipulates them for different types of statistical analysis.

The Report Generator module consists of report generation capability using data from the responses from the Statistical Analysis Module. This module is being developed.

CONCLUSION

The system has been designed and developed to serve as an effective tool to support decision-making tasks of library administrators in their day-to-day management and network operation. The system is currently at the exploratory stage. A number of papers are to be published showing how this system has been used in solving some of the major problems that library administrators confront in the management of library and resource-sharing network operations. Some of the types of questions that this system will help answer have been anticipated as the following:

- In designing a network, which configuration of members will result in the fastest service and the most economical service?
- If the library administrator plans to change the mode of processing data from a manual system to a computerized system, what will be the costs of processing and the number of personnel necessary?
- If the demands on any library are increased by 50 percent in one area, e.g., interlibrary loan, what will be the effect on other areas, e.g., local circulation or the costs of delivering materials?
- If the library administrator institutes a policy of using interlibrary loan for a relatively great number of journals instead of acquiring them locally, what will be the impact on the library's own in-house usage, on the network load of the libraries that the library borrows from, on the photocopying costs?
- If the funding is increased by 20 percent, what will be the impact on service offered, number of transactions processed, percent of demand fulfilled?

These questions are merely representative. It is anticipated that the system will permit the library administrator to answer a host of questions not anticipated here.

With the help of this system, the library administrator can examine several alternatives by asking what-if questions and can make decisions by the comparison of alternatives and the evaluation of outcomes.

A brief manual is under preparation to help users to operate the system. The system is located in the University of Pittsburgh computer system and the user will dial in using a terminal, a modem, and a telephone.

The system is user friendly. The EDIT module initializes the run of the Simulator. It contains help commands that let the user proceed through successive layers of acceptable commands. However, at the initial stage the user will need the manual to know how to log in to the system online, to log off, and to issue commands while using the system.

For the benefit of those library administrators that cannot access the Pitt Simulator system online, a number of critical tables are to be developed based on the simulation of data concerning various library and network operations.

REFERENCES

1. Chris Argyris, "Organizational Learning and Management Information Systems," *Accounting Organizations and Society* 2:113–23 (Nov. 1977).
2. Henry Mintzberg, *The Nature of Managerial Work* (New York: Harper, 1973).
3. G. R. Wagner, "Optimizing Decision Support Systems," *Datamation* 26: 209 (May 1980).
4. Peter G. W. Keen, "Decision Support Systems: A Research Perspective," in *Decision Support Systems: Issues and Challenges*, eds. Goran Fick and Ralph Sprague (New York: Pergamon, 1980), p.65–66.
5. Robert J. Thierauf, *Decision Support Systems for Effective Planning and Control: A Case Study Approach* (Englewood Cliffs, N.J.: Prentice-Hall, 1982).
6. Robert H. Bonczek and others, *Foundations of Decision Support Systems* (New York: Academic, 1981), p.12.
7. Jay W. Forrester, "Managerial Decision Making," in *Management and the Computer of the Future*, ed. Martin Greenberger (Cambridge, Mass.: MIT Pr., 1962), p.37.
8. Marshal C. Yovits and others, "Information Flow and Analysis," *Journal of American Society for Information Science* 32:187–210 (May 1981).
9. Jay Galbraith, *Organization Design* (Reading, Mass.: Addison-Wesley, 1977) p.33–39.
10. Philippe J. Dumas, "The Management of Information Systems: An Information Processing View of Its Environment," in *The Information Systems Environment*, ed. Henry C. Lucas (New York: North Holland, 1980), p.73.
11. A Bosman and H. H. Sol, "A Simulation Based Inquiry System for the Development of Information Systems," in *The Information Systems Environment*, ed. Henry C. Lucas (New York: North Holland, 1980) p.257–73.
12. Allen Newell and Herbert A. Simon, *Human Problem Solving* (Englewood Cliffs, N.J.: Prentice-Hall, 1972).
13. Peter G. W. Keen and Michael S. Scott Morton, *Decision Support Systems: An Organizational Perspective* (Reading, Mass.: Addison-Wesley, 1978), p.65–66.
14. Michael S. Scott Morton, *Management Decision Systems* (Boston: Harvard Univ., Graduate School of Business Administration, 1971).
15. G. Anthony Gorry and Michael S. Scott Morton, "A Framework for Management Information Systems," *Sloan Management Review* 13:55–70 (Fall 1971).
16. Herbert A. Simon, "What Computers Mean to Man and Society," *Science* 195:1186–91 (Mar. 1977).
17. Robert N. Anthony, *Planning and Control Systems: A Framework for Analysis* (Cambridge, Mass.: Harvard Univ., Graduate School of Business Administration, 1965).
18. Herbert A. Simon, *The New Science of Management Decision* (New York: Harper, 1960).
19. Henry Mintzberg, "The Myths of MIS," *California Management Review* 15:92–149 (Fall 1972).
20. Keen, "Decision Support Systems," p.23–41.
21. Allen Kent, "Network Anatomy and Network Objectives," in *Structure and Governance of Library Networks*, eds. Allen Kent and Thomas J. Galvin (New York: Dekker, 1979), p.3–18.
22. William R. Rouse and Sandra H. Rouse, "Analysis of Library Networks," *Collection Management* 3:139–49 (Summer/Fall 1979).
23. H. E. Broadbent, "Evaluation of Resource Sharing in Networks and Cooperatives: A Literature Review for Managers," *Resource Sharing and Library Networks* 1:30 (Summer 1982).
24. James G. Williams, "Performance Criteria

and Evaluation for a Library Resource Sharing Network," in *Library Resource Sharing*, eds. Allen Kent and Thomas J. Galvin (New York: Dekker, 1977), p.227.

25. A. Tversky and D. Kahneman, "Judgement Under Uncertainty: Heuristics and Biases," *Science* 185:1129–31 (Sept. 1974).
26. Allen Kent, "Crystal Gazing into the Future," *Journal of Library Automation* 11:329–37 (Dec. 1978).
27. James G. Williams and others, "Simulation of Information Networks for Studying the Economics of Resource Sharing," *Proceedings of the 44th American Society for Information Science Annual Meeting* (White Plains, N.Y.: Knowledge Industry, 1981), p.379.
28. Robert F. Shannon, "Simulation: A Survey with Research Suggestions," *AIIE Transactions* 7:289–301 (Sept. 1975).
29. Steven Alter, *Decision Support Systems: Current Practices and Continuing Challenges* (Reading, Mass.: Addison-Wesley, 1980), p.73–91.
30. Allen J. Heindel and H. Albert Napier, "Decision Support Systems in Libraries," *Special Libraries* 4:319–27 (Oct. 1981).
31. Ronald W. Chorba and Michael R. W. Bommer, "Developing Academic Library Decision Support Systems," *Journal of the American Society for Information Science* 34:40–50 (Jan. 1983).
32. Jerry W. Mansfield, "Using an INS Simulation to Forecast Manpower Needs," *Journal of the American Society for Information Science* 33:217–22 (July 1982).
33. W. M. Shaw, Jr., "Computer Simulation of the Circulation Subsystem of a Library," *Journal of the American Society for Information Science* 26:271–79 (Sept.–Oct. 1975).
34. Rouse and Rouse, "Analysis of Library Networks," p.140.
35. Allen Kent and others, *Use of Library Materials: The University of Pittsburgh Study* (New York: Dekker, 1977).
36. K. Leon Montgomery and others, "The Economics of Information Transfer Using Resource Sharing Networks: Network Modeling Simulation" (School of Library and Information Science, Univ. of Pittsburgh, Pittsburgh, Penn., 1982 Mimeographed).

Resolving Conflicts in MARC Exchange: The Structure and Impact of Local Options

Robert Renaud

The existence of formats for machine-readable cataloging that use differing sets of content designators entails conflict in the exchange of bibliographic information. Using a basic vocabulary and simple model, this paper discusses four options available at the level of the individual bibliographic database for resolving conflicts in MARC exchange: addition, deletion, modification, and superimposition. The costs and benefits of each option are discussed along with their impacts on MARC exchange.

The existence of formats for machine-readable cataloging that use differing sets of content designators entails conflict in the exchange of bibliographic information. Yet the nature of this conflict and how it is resolved receives only occasional and indirect attention in the literature.[1] This paper discusses the options available to database managers when MARC exchange formats defined by national libraries and formats defined for records in bibliographic databases conflict. In order to facilitate discussion, a basic vocabulary and a simple model are provided. Using this model, four options are discussed. Finally a series of conclusions and suggestions for further research are made. The paper seeks to answer a number of questions: What options are available to database managers when formats conflict? What are some of the costs and benefits of these options? What effect do these options have on the exchange of bibliographic information?

A BASIC VOCABULARY

Before presenting a model of machine-based bibliographic exchange, it is necessary to define the context in which exchange takes place (the library network), the types of machine-readable cataloging found in the library network, and the elements common to all machine-readable cataloging records. This section draws substantially on the work of Attig and Weisbrod.[2]

The context in which machine-based bibliographic exchange takes place is the library network. For the purposes of this paper, the library network will be said to consist of five levels: international, national, intranational, regional, and local. At the international level, agreements between national libraries facilitate the exchange of bibliographic information. At the national level, institutions such as the Library of Congress, the British Library, and the National Library of Canada provide leadership, coordinate programs, and collect library materials within national boundaries. Organizations such as OCLC in the United States and UTLAS in Canada provide library services at the intranational level while remaining independent of national libraries. Regional organizations are groups of libraries located in specific geographic regions that exist to share library resources or services. For example, PALINET is a network of libraries located

Robert Renaud is with the Metropolitan Toronto Library Board.

in Pennsylvania sharing cataloging support services provided by OCLC. The local level of the library network consists of individual institutions such as school, public, and special libraries.

Institutions at all levels of the library network create machine-readable cataloging information for their own use and for the purposes of exchange. The basic unit of machine-readable cataloging is the record. Records are units of cataloging information in machine-readable form that relate to distinct library materials such as books, serials, or films. A collection of records created and maintained by a library organization constitutes a database.

In order to describe the flow of records through the library network, it is necessary to define four types of machine-readable cataloging: MARC exchange, non-MARC exchange, internal, and display. As their names suggest, the first two types of machine-readable cataloging serve to exchange bibliographic information between databases. The term *MARC* refers to all machine-readable cataloging conforming to the international standard for bibliographic exchange (ISO 2709) or its national equivalent (ANSI Z39.2-1979). UNIMARC, LC MARC, and OCLC MARC records conform to these standards of MARC exchange. Non-MARC exchange records are simply records created for the purposes of exchange that do not conform to these standards. Both MARC and non-MARC exchange records move bibliographic information from one database to another. In contrast, internal records are units of bibliographic information residing within databases. For example, records residing in the UTLAS database are called LHF3 (Library Holdings Format, third edition) records.[3] The last type of record is the display. This record represents how an internal record appears to most users of a database. For example, most databases allow users to edit records online. These records appear to the user in a form convenient for examination and editing on a terminal. Here database software restructures the internal record so that users may manipulate machine-readable cataloging with ease.

Although these types of record serve different purposes, they all possess the same basic elements: structure, content designators, and content. The structure of a record refers to its overall organization. For example the structure of all MARC exchange records consists of a leader, a record directory, control fields, and variable fields. Content designators are characters or symbols that serve to identify explicitly types of bibliographic information within records. In MARC exchange records, content designators are tags and indicators at the field level and codes at the subfield level. For example, the characters "100" are a tag for the type of information, "Main Entry—Personal Name," in LC MARC records. The third element of records is the content or data. Content may consist of coded data in control fields or free-form data in variable fields.

Sets of content designators exist to identify the content of records residing in and moving between databases. These sets of content designators are called formats. For example, the LC MARC format consists of a set of coded values, tag values, and subfield codes that identify the contents of LC MARC records. This paper specifically addresses how conflicts between MARC exchange formats and internal formats, that is between varying sets of content designators, are resolved.

Figure 1 relates the levels of the library network and the types of record that have been described. In attempting to understand the flow of records within the library network, it is important to keep in mind the multidirectional pattern of exchange. Whereas MARC exchange records once flowed from the top down, from national libraries to intranational, regional, and local databases, now both MARC and non-MARC exchange records move between databases, from bibliographic utilities to local systems, across national boundaries, and so on.[4]

A SIMPLE MODEL

In order to discuss the options available to database managers when MARC exchange and internal formats conflict, a simple model will be defined. The model includes a MARC exchange format defined at the national level (NEF), an internal format (IMF), and a MARC exchange format

LEVEL	TYPE			
	MARC EXCHANGE	NON-MARC EXCHANGE	INTERNAL	DISPLAY
International	X			
National	X		X	X
Intranational	X		X	X
Regional	X	X	X	X
Local	X	X	X	X

Fig. 1. Levels of the Library Network and Types of Record.

used to communicate internal records (LEF). The NEF consists of content designators A, B, and C identifying types of content T1, T2, and T4, respectively. The IMF consists of content designators A, B, and C identifying types of content T1, T2, and T3, respectively. The LEF consists of content designators A and B identifying types of content T1 and T2, respectively. As various options are discussed, additions will be made to LEF. Figure 2 describes the relationships between the three formats. The flow of information described by the model is from the NEF to the IMF to the LEF, or from left to right in figure 2.

It is evident that content designators A and B identify the same types of content in all three formats. However, a conflict exists between the NEF and IMF in the case of content designator C. In the NEF, content designator C identifies T4 whereas in the IMF it identifies T3. The options discussed below represent various solutions to the problem inherent in this conflict.

Two real examples drawn from experiences at UTLAS help to illustrate this type of problem. In 1982 LC added the fixed field code "n" to its MARC exchange format for books to designate "Surveys of the literature in the subject area."[5] The equivalent fixed field in the UTLAS internal format used "n" to designate "Cases, case notes."[6] At the same time, LC added a variable field tag value "018" to designate "Copyright article—Fee code."[7] This tag value was already used in the UTLAS internal format to designate the INTERMARC control number.[8] As in the model, the same content designators identified different types of content in the MARC exchange and internal formats.

This model is highly simplified and is not intended to represent all possible types of conflict. (For example, the model does not

NEF	IMF	LEF
A (T1)	A (T1)	A (T1)
B (T2)	B (T2)	B (T2)
C (T4)	C (T3)	

Fig. 2. A Simple Model of MARC Exchange.

describe the conflict that arises when different content designators identify the same type of content.) Nevertheless, it depicts a common pattern of bibliographic exchange in the library network in which MARC exchange records created at the national level are converted for use in a database and are in turn exchanged between databases below the national level. The model assumes the existence of software capable of changing content designators at the various stages of exchange.

ADDITION

Adding the conflicting type of content to the IMF would appear to be the simplest solution.[9] In the model, addition entails redefining C in the IMF as identifying T4. In order to facilitate the exchange of records residing in the database, the same change applies to the LEF. Because C no longer identifies T3, all records in the database must be reprocessed to eliminate that type of content. Figure 3 describes the solution represented by addition.

NEF	IMF	LEF
A (T1)	A (T1)	A (T1)
B (T2)	B (T2)	B (T2)
C (T4)	C (T4)	C (T4)

Fig. 3. Local Options: Addition.

Addition has obvious advantages. The program performing the conversion from the NEF to the IMF need not be modified to process T4 in a unique way. Addition also maintains the alignment of the IMF and the LEF with the national standard of content designation inherent in the NEF. At the same time, addition involves serious and occasionally prohibitive costs. By redefining C to identify T4 in the IMF, the type of content identified by T3 is lost. Depending on local circumstances, the type of content lost in the process of addition may in fact be considered more important than that gained by the change. However, reprocessing the records contained in the database to reflect the change represents the greatest cost imposed by addition. In large and frequently accessed databases, the cost of reprocessing all records often makes addition prohibitively expensive.[10]

DELETION

For a variety of reasons, database managers sometimes choose to delete, rather than add, types of content identified by identical content designators.[11] Deleting a type of content obviates the need to reprocess the database to reflect an addition. Deletion also frees databases from carrying types of content considered unnecessary at the local level while at the same time conserving storage space. In the model, deletion involves modifying the conversion software so that T4 identified by C in the NEF is deleted when records enter the database. As a result C continues to identify T3 in the IMF and now identifies T3 in the LEF. Figure 4 describes the solution represented by deletion.

The advantages of deletion respond to the database's short-term needs of conserving space, eliminating unnecessary content in records, and avoiding the costs associated with reprocessing the database. The disadvantages of deletion only become apparent in the long term. As the needs of the database's users change over time, a type of content considered unnecessary at the time of deletion may in fact become critically important. Deletion also causes difficulties when databases exchange records. By using a content designator present in the national exchange format to identify a purely local type of content, the database reduces the

NEF	IMF	NEF
A (T1)	A (T1)	A (T1)
B (T2)	B (T2)	B (T2)
C (T4)	C (T3)	C (T3)

Fig. 4. Local Options: Deletion.

usefulness of the records it exchanges within the library network. Figure 4 makes the impact of deletion on the database's MARC exchange format explicit: C identifies T4 in the NEF whereas it identifies T3 in the LEF.

MODIFICATION

The options described up to now entail the loss of bibliographic information. Addition involves the loss of a purely local type of content whereas deletion eliminates a type of content identified in the national format. The option of modification allows the database to retain both types of content by modifying the values given to content designators in the internal format.[12] Figure 5 describes a solution involving modification. In order to carry T4 identified by C in the NEF, the database defines a new value D for this type of content. C now identifies T3 in the IMF whereas D identifies T4. When the database exchanges MARC records, the conversion software deletes T3 and uses C to identify T4. This option allows the database to retain both T4 and T3 while at the same time maintaining the alignment between the NEF and IMF.

NEF	IMF	LEF
A (T1)	A (T1)	A (T1)
B (T2)	B (T2)	B (T2)
C (T4)	C (T3)	C (T4)
	D (T4)	

Fig. 5. Local Options: Modification.

Of the options described so far, modification serves the needs of the database and the library network most efficiently. (In fact modification resolved the conflicts described earlier at UTLAS.) The disadvantages of modification consist mainly of the costs associated with conversion software and storage space. Software used to modify content designators at the various points of exchange requires greater care, maintenance, and documentation than software dedicated simply to adding and deleting types of content.

SUPERIMPOSITION

The fourth option available to database managers, superimposition, is a variant of the option of addition. Addition resolved the conflict in content designation in favor of the MARC exchange format and necessitated reprocessing the database to reflect this change. However, as has been noted, the cost of reprocessing entire databases is frequently prohibitive. Because of this, the managers of databases sometimes add different types of content identified by identical content designators without any further action. Figure 6 describes the effect of this option. C identifies T4 in the NEF and T3 in the IMF. Superimposition involves adding T4 to the database without any further action. Therefore C identifies both T4 and T3 in the IMF and the LEF. T4 becomes, in effect, superimposed on T3 within the database.

The effect of superimposition is to create layers or strata of types of content within the database identified by identical content designators. In the model the strata created by superimposition are represented by T4 and T3 in the IMF. D. Kaye Gapen noted this effect in a paper published in 1977, calling the strata a "patchwork quilt" of differing types of content.[13] In the context of the library network, the effect of superimposition on exchange constitutes the greatest disadvantage of this option. C now

NEF	IMF	LEF
A (T1)	A (T1)	A (T1)
B (T2)	B (T2)	B (T2)
C (T4)	C (T3, T4)	C (T3, T4)

Fig. 6. Local Options: Superimposition.

identifies T4 and T3 in the LEF. Alternatively, C no longer uniquely identifies any type of content. Databases receiving MARC records in the LEF therefore face a new and intractable problem when they encounter this content designator.

CONCLUSION

The model used in this paper simplifies a process that is in reality difficult and complex. Nevertheless the options of addition, deletion, modification, and superimposition describe how all conflicts in content designation between MARC exchange formats and internal formats are resolved. It is important to remember that one of these options must without exception be selected by a database when such conflicts occur. Even a decision to do nothing, to allow a conflicting type of content to enter the database without further action, results in an unintended decision to superimpose. Having established this, it becomes evident that resolving conflicts in MARC exchange always involves costs and benefits. Addition retains a type of content identified in the MARC exchange format at the expense of a more local type of content. Deletion retains a type of content considered important at the database level while eliminating one identified in the national MARC exchange format. By adding a content designator to the internal format, modification allows the database to retain both types of content. However, modification adds to the costs associated with conversion software. Allowing conflicting types of content to enter the database without any further action reduces software costs but results in superimposition.

The impact of these options extends to the exchange of MARC records between databases. In practice, MARC exchange formats defined below the national level adhere as closely as possible to the standard of content designation inherent in the national exchange format. Like UNIMARC at the international level, the national exchange format acts as a kind of common denominator of bibliographic exchange between databases. When other MARC exchange formats vary, that is, when they diverge from the national exchange format, conversion software must be modified and the cost of bibliographic exchange increases. The model illustrates the impact of various options on the alignment of MARC exchange formats defined at the national and subnational levels of the library network, or the NEF and the LEF. Addition and modification maintain the alignment of the NEF and the LEF by retaining T4. However, deletion and superimposition draw the LEF out of alignment with the NEF by causing either the deletion or incomplete identification of T4. By drawing MARC exchange formats out of alignment with the standard of content designation inherent in the national format, deletion and superimposition add to the cost of bibliographic exchange.

While it is possible to classify the available options and examine their impact, a number of crucial questions remain. To what extent do MARC exchange and internal formats actually vary? What factors influence the selection of options at the subnational levels? What marginal costs are imposed on bibliographic exchange by variations among MARC exchange formats? These questions point out the paucity of our knowledge of the context of bibliographic exchange. As D. Kaye Gapen has written, we need to understand in detail the internal formats and design constraints of existing databases.[14] However, whereas many questions remain unanswered, the model developed in this paper makes evident the need to avoid those options that complicate and add to the cost of MARC exchange within the library network.

REFERENCES

1. Ruth Irvine, "MARC Tagging Structure: Local Variations at Southampton University Library," *Program* 6:286–94 (Oct. 1972); Paula C. Goossens, Jaklien M. Bourgois-Derison, and Andre B. Hubert, "The Internal Processing Format for Bibliographic Data at the Royal Library of Belgium," *Journal of the American Society for Information*

Science 27:147–52 (May–June 1976); Harold Dierickx, "Toward a Common International Bibliographic Exchange Format? The International Symposium on International Exchange Formats, April 1978," *International Cataloguing* 7: 19–24 (Apr.–June 1978); D. Kaye Gapen, "MARC Format Simplification," *Journal of Library Automation* 14:286–92 (Dec. 1981).

2. John Attig, "The US/MARC Formats: Underlying Principles," *Library of Congress Information Bulletin* 41:120–24 (Apr. 1982); Library of Congress, *Principles of MARC Format Content Designation* (Washington, D.C.: LC, 1981). Draft version based on a report prepared by David Weisbrod.
3. UTLAS, Inc., Format For Standardized Bibliographic Records, UTLAS Technical Document no.12 (LHF3) (Toronto: UTLAS, 1983).
4. Henriette D. Avram and Sally H. McCallum, "Directions in Library Networking," *Journal of the American Society for Information Science* 31:438–44 (Nov. 1980); "New Study of MARC Formats Planned," *Library of Congress Information Bulletin* 38:462–64 (Nov. 1979).
5. Library of Congress, Automated Systems Office, *MARC Formats for Bibliographic Data* [*MFBD*] (Washington, D.C.: LC, 1980), 008/24(2), p.1.
6. LHF3, p.34.
7. MFBD, 018, p.1.
8. LHF3, p.104.
9. Irvine, "MARC Tagging Structure," p.288–94.
10. Gapen, "MARC Format Simplification," p.289.
11. Irvine, "MARC Tagging Structure," p.288–94.
12. Ibid.
13. Gapen, "MARC Format Simplification," p.289.
14. Ibid., p.291.

■■

Authority Control in the Online Environment

Lorene E. Ludy and Sally A. Rogers

Authority control work has been affected as libraries implement online catalogs. Individual libraries rely on the Library of Congress but still need local control for original cataloging and for integration of old headings with new or changed headings. Automation allows showing the work of establishing and maintaining headings through cooperative ventures like the NACO project.

Methods of providing authority control have changed dramatically in recent years in response to library automation. In his review of cataloging in 1982, Gordon Stevenson states that the local authority file is becoming obsolete.[1] Yet as long as individual libraries have local catalogs, the function of authority control—to ensure consistency of headings—needs to be reflected at the local level.

Authority work changed at the Ohio State University Libraries (OSUL) as cataloging became automated and control was provided for an online catalog. The changes have preserved the traditional functions of authority control, while making use of the capabilities of the computer and the availability of machine-readable data. Shifts have occurred in when, how, and by whom authority work is done. These shifts can be described as the move from pre- to postcataloging authority work; the separation of mechanical and intellectual tasks, with the automation of the former; and the imposition of authority control at different levels (i.e., national library, network, or individual library).

Before automation, authority work was done at the time of cataloging. All the headings to be used were checked in the authority file. This checking revealed whether the headings had been used: if so, in what form; if not, whether they were variant forms of existing headings or new to the catalog. Headings that were new to the catalog were established: a unique form (distinct from other headings in the catalog) was formulated, variant forms identified, and the relationships between headings shown.

Auld points out that the availability of cataloging copy (LC cards) caused many libraries to do away with their authority files.[2] This is evidence of the shift in the level of authority work, from the individual library to the Library of Congress. The intellectual work of determining unique forms did not have to be repeated, but it was still necessary to integrate headings into the local catalog (the element of control). For this reason, OSUL continued to maintain an authority file. Authority work was done at the time of cataloging, but became primarily the recording of LC's decisions and establishing references for the local catalog.

The use of machine-readable cataloging data further modified authority work. At OSUL, the decision was made to increase cataloging productivity by removing au-

Lorene E. Ludy is authorities librarian and Sally A. Rogers is general cataloger at Ohio State University Libraries, Columbus.

thority work from the automated (copy) cataloging procedure. The authority work that had been done before cataloging was now done after the catalog cards were received from OCLC. The main-entry card was used to check all headings against the authority file and to establish new headings for the file. Authority work also included the correction of headings that were not consistent with the catalog (for example, they matched a *see* reference, were different forms of name, etc.). This separation of authority work from the cataloging procedure was a division of labor in response to the online environment.

Another division of labor also developed: the separation of the mechanical and intellectual tasks of authority control. Matching headings is not an intellectual task; solving discrepancies and establishing new headings are. However, by turning heading matching into a purely mechanical task, the control function of integrating new with existing headings in an individual catalog is lost. While mechanical matching determines that the heading coming into a catalog is already present, it cannot determine whether the new usage is consistent with the old. The abandonment of local control allows productivity to increase; the work done elsewhere is accepted without further evaluation. It is based on this assumption: the library's catalog is a subset of a larger catalog (i.e., the Library of Congress'), so if this larger catalog is consistent, it is not necessary to check for consistency at the local level.

Authority control provides the structure necessary for a catalog by distinguishing headings and showing relationships between and among them. The online catalog, therefore, has to include not only machine-readable records but also machine-readable equivalents of references. In planning the evolution of our online catalog, we included the capability to store and display references. This was achieved through the creation of a headings file that serves as an authority file.[3] The file controls the catalog by serving as an index to the bibliographic records. Additional features are the ability to make global changes to headings and the automatic matching of incoming headings against existing ones.

Ideally, authority work is exercised concurrently with the establishment of the catalog. At OSUL, however, the machine-readable records (the basis of the online catalog) were created before the capability to control them existed. It was especially important to go back and establish control in two areas: to consolidate variant forms under a unique heading and to add references.

When the headings file was created from the headings in the bibliographic records, the precision of machine matching distinguished minor differences (see figure 1). A year-long editing project consolidated variant forms of names. Problem headings were identified both in normal work flows and through the systematic checking of frequently used names (literary authors, composers, etc.). Two librarians and one experienced support-staff member (each working part-time on the project) resolved conflicts and prepared correction forms. A half-time clerk typed optical scanning forms for offline input. Most of the reported problems involved typographical errors or the addition or omission of information such as a middle name, dates of birth and death, etc. A few other discrepancies resulted from differences in spacing, punctuation, or word order. In all, variant forms were consolidated for almost four thousand headings. Figures 1 and 2 show the headings file before and after this editing. The AACR2 form of name was used if available.

Unique headings are only part of authority control. The other part is the references. The availability of the Library of Congress' Name Authority Tapes allowed for the addition of references using that machine-readable data. Records from these tapes were added to our online catalog. A record was added when a heading on the tape matched a heading in our catalog; the references associated with the name were added. Altogether, more than 175,000 references were added.[4] An added benefit of using the Name Authority Tape was the ability to flip to the AACR2 form of heading when the earlier form was present in

```
 1 BENSON, DOUGLAS K.
13 Benson, E. F.
30 BENSON, EDWARD FREDERIC
 3 Benson, Edward Frederic,  1867-
54 Benson, Edward Frederic,  1867-1940.
 1 BENSON, EDWARD FREDERIC, 18671940.
 1 BENSON, EDWARD FREDERIC. U867-1940
 1 BENSON, EDWARD FREDERIKD1867-1940.
 1 BENSON, EDWARD FREDERICK
 3 BENSON, EDWARD FREDERICK. 1867-1940
 1 BENSON, EDWARD WHITE, BP. OF CANTERBURY
```

Fig. 1. The Headings File as It Was Created.

```
  1 BENSON DOUGLAS K.
107 Benson, Edward Frederick,  1867-1940.
  1 BENSON, EDWARD WHITE, BP. OF CANTERBURY
```

Fig. 2. The Headings File after Editing.

our catalog (similar to OCLC's use of the tape in December 1980). Figure 3 shows such a flip. More than 24,000 headings and seventy-two thousand associated records were so changed.

If the OSU Libraries' catalog had contained only Library of Congress records since 1978 (when machine-readable authority records were first created) or only frequently used headings (which were retrospectively converted), the application of the Name Authority Tape would have been the ideal way to bring the local file under control with no human intervention. In reality, our catalog contains nearly one hundred years of cataloging. Only 20 percent of the names in the catalog matched names on LC's tapes. Because portions of the data in the catalog had been added without authority control, some of the changes imposed by the tapes were not valid, or were at least suspect, and required checking by a cataloger. Approximately 4 percent of the changes were determined to be erroneous and required correction.

The use of the Name Authority Tape brought part of the existing file under authority control; these headings were given a verified status. The remaining headings will be controlled as they are used again. When new bibliographic records are added to the catalog, their headings are automatically matched against the headings file. Those that are not found or that match unverified headings are reported. These are searched in the LC Name Authority File on OCLC, and references added manually when found. The remaining headings are evaluated by a cataloger. While the matching functions of authority control have been simplified and speeded up by automation, the reasoning functions are left to the human intellect.

This procedure utilizes the authority work done at a higher level, but only begins to tap the advantages of automation. Data

```
  1 BENSON, DOUGLAS K.
107 Benson, E. F. (Edward Frederic),  1867-1940.
    Benson, Edward Frederic,  1867-1940.
      SEE  Benson, E. F. (Edward Frederic), 1867-1940.
  1 BENSON, EDWARD WHITE, BP. OF CANTERBURY
```

Fig. 3. The Headings File after the Application of the Name Authority Tapes.

is transferred manually between two machine-readable files of authority records. This transfer is necessary because we want the references established by LC to appear in our catalog. While the addition of the Name Authority Tape to the catalog provided onetime addition of references, this is not considered a viable alternative for ongoing authority control. The addition of future tapes would have to be delayed until the headings on the tapes had been used in our catalog; thus there is a trade-off between convenience and timeliness. Because only 26 percent of name headings have references,[5] the processing of the tapes and programming support is deemed too expensive for us.

An alternative method would be to compare the catalog headings to the authority file, rather than the authority file to the headings. This would have to be done for us by a network or consortium as it is not feasible for the OSU Libraries to maintain the LC Authority File to control incoming data. The University of California system uses this method. When a library uses a controlled database, the authority work is concentrated and does not have to be duplicated by each member library. A database with no control of headings in incoming records could offer control of outgoing records. The headings used could be matched against the Name Authority File and a library's archive tape could include both cataloging and authority records.

The preceding discussion of authority work shifting to a higher level, outside the individual library, does not take into account original cataloging. The assumption that a library's catalog is a subset of a larger catalog is true only for a library that catalogs only with copy. A library's catalog is not a subset of a union catalog (such as OCLC) until the unique titles cataloged by that library are added to the union catalog. What is the impact on authority work?

We have found that the authority work done by the catalogers has not changed with automation. It is still done prior to cataloging and involves checking proposed headings against existing ones. New headings are established, with variant forms and related headings identified. However, authority work is done against the union catalog: proposed headings are checked against the local catalog, the LC Name Authority File, and the OCLC database. Our catalogers no longer integrate new headings only into the local catalog; they integrate them into OCLC.

These, then, are the paradoxes of authority work in an online environment. Automation enables resource sharing that allows authority work to be done at a higher level. The individual library no longer does all its authority work. At the same time, the authority work that is being done at the local library level is more complex, time-consuming, and costly because of the demands of resource sharing. This work is not shared and is repeated by many libraries.

Ideally, any authority work done on the local level should be made available to other libraries to reduce duplication of effort. A step in this direction is being taken by the Name Authority Cooperative (NACO) project. Selected libraries create authority records that are added to LC's Name Authority File, making those libraries' authority work widely available. Libraries that gladly avoid the cost of total local authority control by accepting the "catalog subset theory" must realize their role in creating, through original cataloging, the universe of cataloging. Original cataloging must be consistently integrated into the catalog (database) through authority work.

Automation presents another challenge to authority control. The compatibility of machine-readable records has led to the expansion of local catalogs with previously unavailable data, which is often not under control. Our online catalog, for example, also contains records for ERIC documents and acquisitions records. At the same time that we are working to bring our catalog under control, we are routinely adding uncontrolled data to the catalog. We are facing the reality that the consistency of authority control is not always compatible with the desire to provide more information.

Avram states that when authority work takes place higher in the hierarchy, there is less duplication of effort.[6] Automation provides another intriguing alternative: authority work can be shared regardless of the level at which it is created. The authority work of an individual library can be contributed to a consortium or network. And the authority work of the user can be integrated into a catalog.

Part of the purpose of authority control is to free the user from having to identify distinct forms of headings when searching the catalog. The variations are identified at a different level and references provided. The user, however, may have variant forms unanticipated by the creator of the references. An automated system can record headings used in unsuccessful searches, which can be added to the catalog (as references) if necessary.

Authority work has changed and continues to change in the online environment. The prediction of Nancy Williamson is already holding true: "The intellectual challenges of organizing information and retrieving it will not disappear. . . . Authority files could assume an increasingly important role as control mechanisms, although their format and content may be significantly different from the authority files with which we are currently familiar."[7] Perhaps the individual authority file as a passive tool has become obsolete. Online authority files are interactive: they can interact with the catalogs that they control and with the users of those catalogs. Authority work may be done at different levels, in differing degrees. Adapting authority work to control online catalogs remains a challenge to librarians.

REFERENCES

1. Gordon Stevenson, "Descriptive Cataloging in 1982," *Library Resources & Technical Services* 27:259–68 (July/Sept. 1983).
2. Larry Auld, "Authority Control: An Eighty-Year Review," *Library Resources & Technical Services* 26:319–30 (Oct./Dec. 1982).
3. Lorene E. Ludy and Susan J. Logan, "Integrating Authority Control in an Online Catalog," *Proceedings of the ASIS Annual Meeting* 19:176–78 (1982).
4. Lorene E. Ludy, "LC Name Authority Tapes Used by Ohio State University Libraries," *Information Technology & Libraries* 3:69–71 (Mar. 1984).
5. Lucia J. Rather, "Authority Systems at the Library of Congress," in *Authority Control: The Key to Tomorrow's Catalog*, ed. M. W. Ghikas (Phoenix: Oryx Pr., 1982), p.158–65.
6. Henriette D. Avram, "Authority Control and Its Place," *Journal of Academic Librarianship* 9:331–35 (Jan. 1984).
7. Nancy J. Williamson, "Is There a Catalog in Your Future?" *Library Resources & Technical Services* 26:122–35 (Apr./June 1982). ■■

Special Section: Retrospective Conversion

Editor's note: Conversion of library records from manual to machine-readable form has dominated the activity of many libraries as they move toward an automated circulation system or an online or COM catalog. This section describes the methods and procedures used for retrospective conversion for three different systems. The first article discusses the effort in Wisconsin to create a statewide bibliographic database for all types of libraries. The second article deals with the retrocon process for the LCS circulation/resource-sharing system in Illinois academic libraries. The third outlines a project for retrospective conversion with REMARC to build an online catalog. The fourth and final paper focuses on authority control in the retrocon process.

MITINET/retro in Wisconsin Libraries

Robert Bocher

INTRODUCTION

The Wisconsin Department of Public Instruction, Division for Library Services, has developed a statewide bibliographic database and union COM catalog of 1.8 million titles and 5.7 million holdings contributed by Wisconsin libraries. Although most of the records in the database were cataloged by libraries through OCLC, there was an obvious need to develop a method that would allow other libraries in the state to contribute titles and holdings to the statewide database. The MITINET/retro ("mighty-net") system: (1) allows all types of libraries to participate in the statewide database; (2) facilitates retrospective conversion to the MARC format. The following article briefly outlines the development of the Wisconsin bibliographic database and in more detail explains the development of MITINET and its current use in libraries throughout the state.

Robert Bocher is manager, State Bibliographic Database, Division for Library Services, Wisconsin Department of Public Instruction, Madison.

BACKGROUND

Wisconsin developed an official policy statement on library automation, which was released by the state Division for Library Services (DLS) in March 1982. The major objectives of the statement include

- developing and maintaining a statewide bibliographic database;
- standardizing bibliographic records following the MARC II format;
- supporting retrospective conversion by awarding LSCA-funded grants;
- developing a microcomputer program to assist libraries with conversion.

In March 1983 the first edition of the state union catalog (WISCAT) was produced from the state database on 42x microfiche. A second edition of WISCAT with 1.85 million titles was produced in March 1984. The catalog was distributed at no cost to libraries that contributed their bibliographic records and to libraries using the MITINET/retro program. WISCAT is used for a variety of purposes including interlibrary loan, cataloging and classification, acquisitions, reference, bibliographies, and retrocon using MITINET. Brodart, Inc., is the vendor responsible for most of the data processing and the production of WISCAT.

THE DEVELOPMENT OF MITINET/retro

Prior to MITINET it was not possible to contribute titles and holdings information to the statewide database unless a library catalogued via OCLC or had locally developed machine-readable records. Most small and medium-size public libraries, special libraries, and almost all school libraries in Wisconsin do not catalog on OCLC. Yet the collections from these sources are important for local and state resource sharing, and their conversion is a necessity for local or regional automation projects. The MITINET/retro system was developed to potentially allow all libraries in the state to participate in the statewide database and at the same time convert their catalogs into the MARC format. MITINET has been a joint cooperative development between the private sector and government. Hank Epstein of Information Transform, Inc., developed the system and wrote the functional specifications. Paul Jakoubek from DLS did the programming. MITINET/retro currently is compatible with most Apple and IBM microcomputers.

MITINET: How It Works

The MITINET system takes advantage of the conversion work already accomplished by the Library of Congress and libraries in the state. This allows libraries to avoid keying in bibliographic information. Retrospective conversion using MITINET involves two basic processes. First, a library receives a copy of WISCAT and attempts to match a title in their collection with the same title on WISCAT[1] by comparing author, title, edition, and other bibliographic information. When a match is found, the unique eight-digit "MITINET" number is taken from the entry on WISCAT and written on the catalog or shelflist card. This number is the key to the specific title matched in the statewide database. After a series of titles have been matched, the MITINET/retro program is booted up.

During the very first use of MITINET, a library creates a "custom" disk. This floppy contains basic library profile information, including a library's name, branches or collection codes, and the primary classification system used. Creating a custom disk is only done once unless at some future date the library needs to change any of the information. The data that are stored on the "custom" floppy are read by the MITINET/retro program and retained for the remainder of the data entry session. Information on the custom disk is used to generate screen formats, which reduces the data needed to input a record and minimizes keying errors.

After a custom disk has been created, the second phase of conversion involves keying in pertinent data from matched titles on the microcomputer. MITINET is menu driven and takes the operator through a series of options often using default prompts that can be overridden if the presumed activity is not desired. After booting the program and custom disk, a "data" disk is inserted into the drive. The first data input is the eight-digit MITINET number that was written on the shelflist or catalog card. The eighth digit of this number is a check digit that makes it almost impossible to enter the incorrect number. If the preceding seven digits are not correctly entered, a friendly message appears warning that the number entered is not correct. The operator can then check the number on the card and reenter it properly. The program will not allow any other data to be entered until the correct MITINET number has been input.

The next data input is the library's local call number, which can be any combination of alphanumeric characters to a maximum length of twenty-five. Copy-level data can be input for titles held at branch locations or specific collections within a library. When the MITINET number and local information have been input for a title, they are written out to the data floppy by pressing "A" (for adding a record to the floppy) and the "RETURN" key.

A title not matched on WISCAT can be searched for a match using a specially produced edition on microfiche of the Library of Congress MARC database. The MARC fiche contains approximately 1.1 million English-language titles and can be used as the primary source for matching if libraries do not have a large union COM. The LC card number with a check digit specially added to each title is used as the key to specific records. When a title is matched on the MARC fiche, the entire bibliographic record along with the library's holdings data is pulled from the MARC database and merged into the statewide database. If a title cannot be matched on WISCAT or the LC MARC fiche, a library can enter it using the MITINET/marc program. This program is currently being field-tested and is expected to be operational in early 1985.

When Wisconsin libraries fill a floppy disk (each holds one thousand records), they send it to DLS in Madison. DLS staff transfer the data from floppy to hard disk at the Madison Academic Computing Center (MACC) on the UW-Madison campus via a regular dial-up phone link at 1200 bps. After verifying that the data have been correctly transferred from floppy to hard disk, the floppies are returned to the libraries and reused. When the online file at MACC approaches one hundred thousand records, it is spun off on tape and sent to Brodart for merger into the statewide database. All transactions for adding, deleting, or changing titles or holdings are interfiled on the tape. Libraries using MITINET in other states send their floppies directly to the vendor.

MITINET: The Testing Process

Formal development and programming of MITINET began in July 1982. Beta testing in eleven schools, public, and special libraries started in October 1982. Valuable feedback was obtained from the testing libraries and many of their suggestions were incorporated into the program and *MITINET/retro User's Manual*.

Several test files were created and transactions were submitted for adding, changing, and deleting holdings and titles from the test files. Invalid transactions were also submitted (e.g., library *A* attempting to delete a holding input by library *B*) to make certain fail-safe logic was fully operational. Finally, after testing was successfully completed, MITINET became operational in June 1983.

PROBLEMS ENCOUNTERED IN THE CONVERSION PROCESS

No project of this magnitude can realistically be implemented without problems. Difficulties have been encountered in two basic areas: (1) problems that are program related; (2) problems related to the manual matching process using WISCAT. Difficulties in the latter have been much more prevalent than in the former.

Program and Data Conversion

There are three basic transactions in the MITINET system: adding (ADL), changing (CHA), or deleting (DEL) a title or holding from the statewide database. A CHA transaction only works when data from the converted titles have already been merged into the state database. It does not work on the supplement file of MITINET records. Depending on the timing of WISCAT production, it was possible that mistakes made in inputting (incorrect call numbers, branch locations, etc.) would not be corrected prior to the production of WISCAT. Even if mistakes could be corrected, it was imperative that a library keep a written log of all mistakes and enter a CHA transaction at some future date after the holdings data were already merged into the state database.

Therefore, it was highly desirable to allow libraries to correct mistakes *after* the record had already been written out to the floppy disk but *before* it was merged into the state database. It has always been possible to revise data before the last step—writing the record to the floppy—is executed. Unfortunately, many inputters were routinely adding the record to the floppy and then reviewing the data on the screen during the five to seven seconds it took to write to disk. It was not an efficient use of time to make inputters spend several seconds reviewing data before the data were added to the floppy and then having to wait an additional five to seven seconds while the data were added to the floppy.

This problem was solved by having a library enter a DEL (delete) for a record after an ADL (add) for the same record was written to the floppy. This delete logically (but not physically) would cancel the original ADL that contained the mistake. The inputter would immediately follow the DEL with another ADL for the same record entering the correct information. On the floppy there would then be three physical records for the same item; an ADL-DEL-ADL. In testing this series of transactions, it was documented that only the last ADL would be correctly processed and added to the state database. This significantly reduced the number of mistakes that had to be documented and eliminated known holding errors from appearing on WISCAT.

Another problem that inputters found disconcerting was the "disappearing" cursor. As a typist was inputting data, periodically and without warning, the cursor would disappear for a second or two and the micro would not accept data entered during this time. This problem was caused by the Apple microcomputer performing "garbage collection," the clearing out of strings from previously entered data. Often, an error in the input data, such as several characters missing from the call number, would only be detected as the inputter was reviewing the data on the screen just prior to or after adding the data to the floppy. It was then necessary to revise the data and enter any missing information. Because the average record takes only thirty seconds to enter, any revising slowed the inputting considerably. If missing characters were discovered after the record was written to the floppy, a DEL-ADL had to be entered (as described in the previous paragraph), which took even more time. Any mistakes caused by a disappearing cursor at least doubled or even trebled the time needed to input a title. Also, it was impossible for inputters to develop a rhythm for inputting because there was no set point at which the cursor would disappear.

This problem was eliminated by modifying the software and specifically programming the Apple to do garbage collection after all the data had been entered and just before the record was written out to disk. This added an extra two seconds to the normal five to seven seconds it took to write data to disk. However, staff using MITINET on a regular basis preferred this brief delay to the frustration and uncertainty previously encountered.

MITINET/retro, as the name implies, was originally designed for *retrospective* conversion, and it is not yet possible for libraries in Wisconsin to enter new acquisitions. The current edition of WISCAT used for matching includes titles through 1983. Therefore, it is necessary for a library to keep a manual file of new (1984) acquisitions that cannot be entered until after production of WISCAT III in early 1985. The holdings from these titles will not appear until WISCAT IV is produced in 1986. Although it would be possible to enter new titles using the MITINET/marc program, most new acquisitions will also be cataloged by libraries in Wisconsin using OCLC or by the Library of Congress. Statistics from the public and school libraries using MITINET reveal that 85 to 95 percent of newly acquired titles have LC card numbers.

Obviously, it is very beneficial to the libraries and to future editions of WISCAT to be able to input current acquisitions. Brodart has agreed to produce cumulative monthly supplements of the LC MARC fiche that can be used for matching new titles. A computer consolidation on this LC supplement file would be run to eliminate duplicate titles before the file was merged into the state database. Another consolidation would be needed to avoid duplicates between the records pulled from MARC and the same titles cataloged and input into the database via OCLC. Libraries using MITINET outside Wisconsin do not have the duplicate record problems to be resolved, and they can enter new acquisitions at this time.

DLS and Brodart are currently working on specifications to ameliorate problems encountered by Wisconsin libraries entering new acquisitions using MITINET. The option is being explored to have libraries use the monthly MARC supplements but run their records against the state database after it has been updated with the quarterly OCLC input. Titles not matched after several updates would then be run against the LC MARC database. It is anticipated that problems associated with duplicate titles will be resolved and libraries in the state will be able to enter new acquisitions by January 1985.

Matching Titles: WISCAT and the Local Catalog

There are several problems in the manual process of retrospective conversion not directly related to the MITINET/retro software. The matching process obviously takes time, and the difficulties with this process are a result of inherent deficiencies present in WISCAT and problems present in the local catalog or shelflist. Most of the problems in WISCAT result from cataloging practices and record selection criteria used by individual libraries in the state cataloging on OCLC.

When matching titles from their local catalog, many librarians found it irritating that duplicate titles were appearing on WISCAT. After investigating several references to specific titles that were claimed to be "duplicates," it became obvious that even though some entries have the same author/title, they are often not the same item. That is, in many instances, the date of publication, edition statement, publisher, place of publication, or physical description differentiates one "duplicate" from another. It is sometimes difficult to explain to librarians who are not catalogers why items with such differences cannot be considered the same. In fact, it is sometimes difficult to explain this to catalogers.

While many apparently "duplicate" titles are in fact not the same, there are still thousands that are genuine duplicate titles. In the initial processing of OCLC archival tapes, bibliographic records with the same OCLC number were eliminated, but if two identical titles had different OCLC numbers they were not eliminated. As a result they appear in WISCAT as duplicate titles. A consolidation of these titles will be run before the production of WISCAT III in early 1985. The consolidation will compare LCCNs and $A of the 245 tag. When two or more records match both comparisons, the LC MARC record (if one is present) will be retained. A program will also be written to eliminate duplicate titles even when an LCCN is not present in two or more records. All holdings data from eliminated records will be appended to the retained record. The MITINET number and OCLC number from the eliminated records will be cross-indexed to the master record that remains. This will make it possible to submit updating transactions that will be properly processed even if the originally inputted record has been eliminated.

Another problem with WISCAT is its current lack of name authority. This is causing problems in the matching process. The simple lack of an author's middle initial on some titles and its inclusion on others may cause different works by that person not to file together. This is especially true for prolific writers and authors with common names. If libraries use the title for matching, this problem is moot, but many libraries are matching by using the author on their shelflist or catalog card. The possibility of running LC Name Authority is currently being investigated. While this obviously would not solve all name problems, it would be a step in the right direction.

Incorrect filing indicators present in the bibliographic record when it was produced or updated on OCLC have proven to be a troublesome problem. Obvious filing problems include titles filed by initial articles and the letter *O* being input as Ø (zero) and vice versa. Much more insidious filing problems are the entries that are randomly misfiled throughout the catalog. For example, a title beginning *The New* . . . is filed near the beginning of the *E*'s because the second indicator in the 245 tag was set at "2" instead of "4." A library trying to match a title that is misfiled will obviously assume it does not exist. Brodart has developed software that will allow an operator to change the filing indicators on the master record. This "clean-up" is currently taking place.

Another major problem encountered in the matching process is common to many libraries that embark on a retrospective conversion project: the condition of the local shelflist is less than ideal. Less than seven of the libraries currently using MITINET have a full-time professional cataloger on the technical services staff. All the local idiosyncratic cataloging practices loosely followed over the years have not made the conversion process any easier. Different procedures were used, rules followed, and steps defined in the cataloging process that produced more discontinuity than continuity. While many larger OCLC libraries have converted to, or reached an accommodation with, AACR2, many smaller libraries have done nothing to incorporate the new rules into existing cataloging practices. Although the shelflist is supposed to be the most accurate representation of what is actually on the shelves, it is at times woefully inaccurate. The conversion pro-

Table 1. Matching and Inputting

Titles matched using WISCAT:	87.1%
Titles matched using LC MARC fiche:*	36.4%
Combined match rate:	91.7%
Average number of titles matched per hour:	61.9
Average number of titles input per hour:	119.1
Combined conversion rate for both manual matching and inputting on the microcomputer:	40.7

*36.4% of the 12.9% *not* found on WISCAT were matched on the MARC fiche. The percentage matched on MARC will be considerably higher for libraries using it as their primary source.

cess is slowed considerably when a staff member has to physically check the item on the shelves to verify the edition, publisher, or other information because these data were not on the card used for matching.

STATISTICS AND USERS' REACTIONS

The following statistics have been compiled directly from libraries using MITINET. Most libraries in Wisconsin are using their shelflists to do the physical matching. The match rate is highly dependent on the type of library using MITINET and, by definition, its collection. The number of titles matched is highest for trade titles and lowest for audiovisual items and specialized collections. The highest conversion rate was 96.86 percent for a public library, while special libraries are in the 60–75 percent range. Many schools using MITINET are matching 82–88 percent of their print materials (see table 1).

The number of titles matched in a given length of time is highly dependent on the condition of the local catalog or shelflist. The number of titles input is dependent on the number of branches or collections that are defined as part of the custom process. Many public libraries do not classify fiction and thus have no call number to enter for these items, which minimizes the keystrokes needed to convert each title. Some libraries are using numeric keypads for fast entry of the MITINET number and the LCCN. Table 2 provides more detailed information from five libraries that are currently using MITINET.

The New Richmond Public Library is using their public catalog for matching. When using its public catalog and matching by author, many items will be found on the same WISCAT author fiche, which limits the need for changing fiche and speeds the matching process. All other libraries listed are using their shelflists for matching. The conversion rate and percentage matched for Luther Hospital is well below average because of its collection and because the two largest medical libraries in the state are on OCLC but have not done retroconversion. Park Falls Public Library hired six additional staff members, borrowed Apple microcomputers from the local high school, and finished their conversion in six weeks during July and August 1983.

The per-title cost for labor is directly dependent on the number of titles input per hour and the wages paid to staff working on the project. The MITINET/retro software, WISCAT, and the LC MARC fiche are supplied at no cost to Wisconsin libraries. Holdings appended to titles matched on WISCAT are added to the state database at no cost. Titles matched and pulled from the LC MARC database cost $0.07 per unique title. This cost is currently being supported by DLS.

The reaction to MITINET by users has been very positive. Few of the libraries had ever been involved in a conversion project,

Table 2. Statistics from Five Libraries Using MITINET/retro

Library	*Titles converted*	*Titles converted per hour*	*Percent matched*	*Cost per title for labor*
New Richmond Public Library	10,328	55.29	94.60	$0.21
Chippewa Falls Public Library	14,065	35.33	92.18	0.18
Rice Lake Public Library	16,745	48.19	88.47	0.148
Luther Hospital Library	1,186	16.99	72.00	0.72
Park Falls Public Library	24,599	38.80	95.10	0.736

and most staff members had no experience with microcomputers. On-site training is provided by DLS, and a MITINET "demo" program disk is sent to prospective users. After a library starts using MITINET, the staff quickly realize that entering data using the software is the easiest part of the conversion process. The most difficult and time-consuming facet is matching titles.

Some librarians are not enthusiastic about MITINET because it is necessary to manually match a title using a microfiche catalog and fiche reader. Unless one is careful, one can easily become enamored with the form and method of conversion and lose sight of the end purpose. By visually confirming that a title on WISCAT (or the MARC fiche) is the same title on the local shelflist or catalog card, librarians can be assured their holdings will not be appended to the wrong record in the database. Many batched retroconversion systems do not offer this assurance. Mismatches only become apparent when the library starts using its records for local purposes.

Several libraries in the state that have locally developed non-MARC bibliographic records are realizing that using MITINET to obtain full MARC records may be less expensive than attempting to upgrade their local records. As part of the statewide database, 997,301 records from locally developed systems have been run against the state database and the LC MARC database to find a match. The LCCN or a short author/ title algorithm was used for the match. Cost analysis reveals that in several instances it would have cost less to use MITINET than it did to "clean-up" the problems associated with machine matching and merging non-MARC records. Using MITINET (or an online system like OCLC) eliminates the problems associated with keying errors and poorly documented local record formats that can plague a machine match.

MITINET/MARC

As mentioned previously, titles that cannot be matched using WISCAT or the MARC fiche can be entered on the microcomputer using the MITINET/marc program now being field-tested. This program is divided into two levels of cataloging. The first level is a basic program that prompts users for bibliographic elements in English-readable statements (i.e., author, title, publisher, etc.), while internally all data are translated into the appropriate MARC tagging structure. The inputter can select from a limited list of optional fields, including an LC subject heading (650) instead of a local heading (690), and added entries by specific type (700, 710, 711, etc.). The second level is designed to allow any legitimate MARC field to be prompted and input. Libraries at either level will be able to customize which fields they want as mandatory and which fields are optional based on the format they are using.

MITINET: LOCAL USE OF RECORDS

Many libraries are using MITINET not only to contribute to the statewide database and WISCAT but also to obtain MARC records for local and regional use. Several libraries, including eleven members of a public library system, plan on producing local or regional COM catalogs to be distributed to many other libraries in their respective areas. Several libraries have investigated shared circulation systems. While these would obviously be minicomputer-based turnkey systems, there is also a great demand for stand-alone microcomputer circulation systems in smaller libraries.

DLS is currently working with several vendors of microcomputer circulation systems to allow libraries converting via MITINET (or OCLC) to use the same bibliographic records in the vendors' micro-based circulation systems. The vendors that are cooperating with DLS include the Follett Library Book Company, which markets the Book Trak System, and Management Systems, Inc., which supports the Circa circulation system. Under this agreement, a library's MARC records would be extracted from the statewide database and pertinent data stripped off, reformatted, and downloaded into the vendors' circulation system. This arrangement has numerous advantages and will allow libraries to

- use a relatively easy and fast method for conversion;
- participate in the statewide database and WISCAT;
- obtain a full MARC record for any fu-

ture use at the local, regional, or state level;

- eliminate the necessity of typing in all the bibliographic data for the local system;
- create a subset of the full MARC record for use in another system;
- have bar codes automatically generated.

MITINET: CURRENT STATUS

By the end of December 1983, a total of 245,628 titles had been converted and merged into the statewide database by thirty-seven libraries using MITINET. Holdings from these titles appear in the second edition of WISCAT, which was produced in March 1984. Currently, there are fifty-seven libraries using MITINET/retro in Wisconsin. They include thirty-nine public, nine school, six academic, and three special libraries. Several libraries are using OCLC for current cataloging and MITINET for retroconversion. Since the beginning of 1984, libraries have been submitting an average of 11,842 titles per week and more than 500,000 titles had been converted by June.

Following the objectives outlined in the library automation policy statement, the Division for Library Services has funded more than $1.4 million in retrospective conversion grants using Library Services and Construction Act (LSCA) funds in the last four years. Additional LSCA conversion grants were awarded in May of this year to another fifty-five libraries to use MITINET. With these libraries and others using MITINET/retro with local funding, the weekly conversion rate is expected to double to more than 20,000 titles per week. With this increase, 60 percent of the retrospective conversion in Wisconsin will be through MITINET with the remaining 40 percent done through OCLC.

CONCLUSION

MITINET is proving to be an effective, reliable, and relatively inexpensive method of adding titles and holdings to the Wisconsin bibliographic database. At the same time, libraries are converting titles into the MARC format for future use at the local or regional level. Most Wisconsin libraries never considered catalog conversion because no cost-effective and easy method for conversion existed and no clear purpose for conversion was foreseen. The statewide database, WISCAT, and MITINET are dramatically changing that viewpoint for many libraries in the state.

REFERENCE

1. Libraries outside Wisconsin can use MITINET by matching titles using the LC MARC fiche. Consult: Hank Epstein, "MITINET/retro: Retrospective Conversion on an Apple," *Information Technology and Libraries* 2:166–73 (June 1983). ■■

Retrocon for LCS in Illinois Academic Libraries

Doris R. Brown

The adoption of the Library Computer System (LCS) in 1978 as a circulation-based resource-sharing system signaled the beginning of massive retrospective conversion projects for many academic libraries in Illinois. Although several of the state's geographic library systems had already implemented the Computer Library Systems, Inc. (CLSI) as their circulation system, the joint committee of the Illinois Association of College and Research Libraries (IACRL) and the Illinois Board of Higher Education (IBHE) opted for LCS because it was viewed as a proven system able to process the high volume of transactions of a large resource-sharing network.[1] The IBHE had already funded development of LCS for campuses of the University of Illinois in Urbana-Champaign and Chicago, so those institutions already had retrospective conversion projects under way. The library at the Urbana-Champaign campus decided to use an outside vendor to convert records for their monographic collection and to keypunch serial records in-house.[2] The Chicago Circle campus library (now University of Illinois at Chicago) chose to convert manual records via OCLC with its own

Doris R. Brown is director of libraries, DePaul University Library, Chicago, Illinois.

staff in order to create a full bibliographic record and to allow access to its holdings by other OCLC libraries in the state.[3] The Medical Center (now University of Illinois at Chicago Library of the Health Sciences) used the same vendor for monographs as Urbana-Champaign, but added serial bibliographical information to LCS from the Union Catalog of Medical Periodicals database and converted serial holdings from 1965 on via the Superwylbur files of LCS.

All academic libraries in Illinois were invited to apply for participation in the IBHE/IACRL pilot project for sharing resources on a statewide basis. Criteria for selection included: at least one multi-campus institution, geographic spread throughout the state, at least one community college, at least one four-year college, at least one university offering the doctoral degree, both public and private institutions, and institutions of various sizes and with various types of program strengths. The one-year Higher Education Cooperation Act (HECA) grants from the IBHE for the fourteen academic libraries selected for initial participation in the expansion of LCS provided fifteen cents for every record to be converted; this figure was derived from the average per-item cost for record conversion by an outside vendor. The grants were awarded for fiscal year (FY) July 1, 1979–June 30, 1980. The size of the grant to each institution was based on the size of the collection to be converted; each library could decide for itself what method of conversion to use, as long as it provided the short bibliographic record (call number, author, title, edition statement, place of publication, date of publication, LC card number, and language code) and holdings information (copy/volume number, location, length of loan period) necessary for the LCS system. Although there had been initial interest in negotiating with a single vendor for data conversion for all libraries, the idea was abandoned after a sampling indicated the individual library requirements were too varied to allow standardization by one vendor.

ELECTRONIC KEYBOARDING, INC.

In discussion regarding the adoption of LCS, it had been thought that size of the collection was the primary factor in selecting an outside vendor or using OCLC inhouse. The larger libraries that received grants in FY 1979–80, therefore, decided to send shelflist records to an outside vendor for conversion to machine-readable form. Founders Memorial Library at Northern Illinois University (DeKalb) had been involved in database conversion since 1973, first with its local system's Data Base Conversion Unit, then combining that service with OCLC, and at last deciding to complete the process by sending the shelflist to Electronic Keyboarding, Inc. (EKI), the same vendor finally used by the University of Illinois at Urbana-Champaign, to have the truncated LCS information keypunched.[4] The procedure used by Northern and all other libraries using EKI was similar to that developed by the library at Urbana-Champaign. Conversion took place on the vendor's premises, with a previously determined number of shelflist drawers sent to the vendor to be keyed according to a schedule established with each institution. Specifications were drafted based on an average number of 106 characters per title (with eight control characters) for the following fields:

a. Author—the first author on the card—personal, corporate, meeting, or uniform title (average twenty-three characters).

b. Short title—title to the first logical break or to the first character of punctuation, depending upon the individual title and which rule provided the most meaningful short title (average forty characters combined with edition).

c. Edition—the statement immediately following title, when it appeared at all, indicating an edition other than the first.

d. Call number—the basic library control number; one call number per title (average twelve characters).

e. Library of Congress card number (average four characters).

f. Holdings and location data—comprising volume number(s), copy number(s), and location code(s) (average eight characters).

g. Place of publication—the first listed place only to be recorded (average six characters).

h. Publication date—one date (indicating a year) identified in the body of the card (average four characters).

i. Serial/language code—flag character *S* for all English-language serials; where the language of the title indicated a foreign language, flag character *SF* was inserted (average one character).

All fields in a record began with the character @, used as the field delimiter whether containing data or blank. The character $ was used to separate the edition subfield from the title and to separate place of publication from the edition subfield or the title if there was no edition. The vendor was instructed to bypass computer-produced shelflist cards from OCLC or from Northern's own system. OCLC tapes were then merged with the EKI product and the local tapes; the only problem occurred when some Northern tapes were misplaced, causing twenty thousand records in the *P* classification to be lost.

Milner Library at Illinois State University (Normal) also opted for EKI, given the large volume of records (545,000 monographs) to be converted and the reluctance to use a vendor who would match local records to MARC records without regard to local cataloging adaptations. Illinois State had already converted periodical records to a machine-readable form for generating a hard-copy periodical directory and was working on an ongoing project to enter data for the complete central serials file. Before serials could be loaded into LCS, however, the serials data had to be reformatted and reprogrammed to convert from a local program that lacked clearly definable fields. Illinois State decided not to use the archival OCLC records for LCS. Instead OCLC cataloging procedures were reworked to facilitate use of OCLC tapes beginning in October 1979, with OCLC records formatted to provide specific copy/volume/location information for the OCLC/LCS interface.

Kankakee Community College (Kankakee), the only LCS grant recipient not a member of OCLC, followed the example of the other Illinois libraries to convert data for LCS with EKI. To add current materials to LCS, KCC buys catalog cards from Baker and Taylor and then adds short bibliographical information from the cards to LCS in the Superwylbur files. Lovejoy Library of Southern Illinois University at Edwardsville, a grant recipient in FY 1980–81, contracted with EKI for conversion of their 412,000 shelflist records, which included short bibliographic information for nonperiodical serials, but not holdings data. The present project to use OCLC to convert Lovejoy Library's fifteen thousand periodicals and to add the holdings of twenty thousand serials titles already converted by EKI began in 1983, after an extensive revision of the OCLC profile to incorporate organizational change and to accommodate the OCLC/LCS interface.

In all institutions, proofreading EKI's computer printouts was tedious and time-consuming for both regular staff assigned to the project and for student or part-time help hired for the task. Motivating staff was always a problem, so the quality of proofreading ranged from good to indifferent. EKI's preliminary proofing identified errors in the program and patterns of keying errors with such an overall acceptable degree of accuracy that selective review of proof copy might have been sufficient.

BLACKWELL NORTH AMERICA

Sangamon State University Library (Springfield) elected to use a MARC-based vendor for full record conversion because a survey of the shelflist indicated an OCLC conversion could not be completed in the one-year grant period. SSU decided that government documents and periodicals needed only the short record and so contracted with EKI for those two types of materials. Blackwell North America (B/NA) was selected over Brodart for conversion of all other materials because B/NA's database was larger than Brodart's and included materials other than just books. The promised two-week turnaround time and twelve-month time frame for completion were also factors in the selection. SSU hired CETA staff to type the bibliographical information for the short LCS record on optical character recognition sheets with IBM Selectric typewriters. The sheets were mailed to B/NA on a weekly basis, with B/NA production to be returned two weeks later for editing by library staff. In addition to staffing problems with the CETA workers, Sangamon State did not receive the service promised by Blackwell in the agreed time. Because of organizational and personnel changes, B/NA was never able to meet the production schedule, so inaccurate information generated by B/NA could not be corrected by SSU before it was loaded into LCS. At the end of the contract period, 40 percent of the monographic shelflist was either not converted or converted inaccurately. Sangamon State still has an ongoing project to correct or convert monographic records via OCLC.

OCLC

Ten of the fourteen libraries receiving IBHE grants for FY 1979–80 undertook retrospective conversion projects through OCLC: Catholic Theological Union (Chicago), Chicago State University (Chicago), DePaul University (Chicago), Governor's State University (University Park), Illinois Institute of Technology (Chicago), Judson College (Elgin), Lake Forest College (Lake Forest), Millikin University (Decatur), St. Xavier College (Chicago), and Triton College (River Grove). OCLC was selected because these institutions wanted to create a full record in machine-readable form to be able to use the data for an online catalog at some future date. Because all were already using the Cataloging Subsystem through ILLINET, practical considerations meant the institutions were committed to OCLC, and staff were familiar with system commands and formats. Some libraries, like Governor's State University, had begun retrospective conversion as early as 1977, shortly after joining OCLC, so grant funding allowed them to increase the output of their retrocon projects. These small to medium-sized libraries decided they would be best served by using OCLC even though they recognized that the inconsistent quality and heavy duplication of the OCLC database would probably slow down searching and record selection. Cataloging records for all Illinois libraries using OCLC were already available on the state archive tape, which the Illinois State Library began receiving in March 1977. In addition, a "now or never" attitude set in when the libraries receiving grants became aware that, although they would have to bear any extra costs for conversion to the full record instead of the LCS short record, it was highly unlikely that there would be additional grant funding for full conversion at any time in the future. Retrocon projects authorized by OCLC bore no charge during FY 1979–80, so the grant monies for retrospective conversion could be used for leasing or purchasing OCLC terminals and for project staff salaries. Lake Forest College and Catholic Theological Union purchased an OCLC terminal; the other libraries leased from one to three terminals for the project. In order to complete the project within the one-year grant period, all libraries hired staff either full- or part-time to update the OCLC records; levels of expertise of staff hired for the project varied from students who had never used a terminal before to librarians with several years of OCLC experience. Because of the time constraints of the funded projects, regular staff were also assigned to work on retrospective conversion, either on a volunteer or departmental conscription basis.

Training programs were devised by each institution with assistance from the Illinois State Library/Illinois OCLC Users Group continuing education workshops on retrospective conversion. Illinois OCLC libraries were sufficiently concerned about the quality of the OCLC database, both for local cataloging and because of the long-range implications for interlibrary loan and resource sharing, that training programs and policies were careful to adhere to the statewide standards recommended by the state OCLC Users Group. In addition to initial training with a written manual describing the steps in the project, some libraries set up a "buddy" system for part-time staff to give them a staff member knowledgeable in OCLC to answer questions and resolve problems immediately. Project staff were first trained in basic terminal operation, searching and tagging on the terminal. They were then given offline exercises to practice their new skills. Further training in selecting and editing records online followed completion of the exercises. The SAVE file allowed trainers to review initial updating. In a few institutions, technical services staff were not involved in the conversion because LCS was viewed as a circulation (i.e., public services) system; consequently training was inadequate and shortsighted, overlooking the long-range implications of just adding holdings symbols on OCLC. In most libraries, high turnover and limited schedules of retrocon project staff made staff training one of the most labor-intensive and frustrating parts of the whole process.

LCS was scheduled for implementation by libraries, other than the University of Illinois, beginning in July 1980. The pressure of time constraints, both to expend the grant and to bring up LCS on the date scheduled for each institution, forced libraries to concentrate on updating materials, primarily monographs, which would circulate through LCS. Periodicals, microforms, audiovisual materials, and reference collections were left until circulating materials had all been updated. This decision also simplified the conversion process for project staff since there tended to be a higher hit rate for such material.

These ten libraries worked from their shelflists, which were first cleaned up to simplify interpretation for project staff not familiar with shelflist idiosyncrasies. Weeding and inventory accompanied the process in those libraries that could accommodate so many projects at once. Most libraries set up three phases in the retrocon process according to levels of staff expertise: (1) exact matches updated by temporary project staff who flagged all questions and problems for more experienced staff; (2) staff more skilled in OCLC and cataloging practice updated variations in the match with OCLC records, complicated holdings, unclear shelflist information, etc.; (3) professional catalogers resolved problems requiring recataloging, reclassification, original entry, etc. Policies and procedures varied from one institution to another in order to accommodate local cataloging practices, but some rules were almost universally applied: (1) ignore OCLC shelflist cards produced after March 1977 since those records were already on the archive tape; (2) match author, title, place of publication/publisher, date, edition, and pagination with the information on the shelflist card—pagination might vary some, but other elements should match; (3) complete all fixed fields for the record selected; use the default value for information not supplied on the shelflist; (4) complete the appropriate call number field and the 049 field for holdings. In at least one institution, however, part-time staff were instructed to match only author and title with the OCLC record; the confusion from mixed-up editions and publication dates for both local circulation and interlibrary loan is still being unraveled. In another library, staff were not told to add multiple copies to the 049 field so the LCS record now has to be corrected whenever additional copies turn up.

Rates of conversion were high (fifteen to thirty per hour) even though OCLC suffered a lot of downtime. Although the ten libraries updated a large volume of records in one year (593,535 total updates), problems arose in those areas where the LCS programming was not completed or fully tested. After conversion projects were well under way, libraries were informed to delete multiple call number fields to guarantee that only the correct one would be transferred to LCS. Use of the 049 field had not been understood in pre-LCS days, so some libraries had used it for accession numbers or other local non–holdings data for material cataloged or updated earlier. Many had just ignored holdings completely since OCLC didn't require them, which meant re-updating materials already on the pre-1979 archive tape or adding holdings directly to LCS later. The format for entering complicated holdings in the 049 field was sketchy at the outset of the projects, developing and changing while libraries were in the process of updating, so staff confusion and errors were inevitable. When the limit of 1,230 characters for the 049 field became apparent, it was solved by dividing holdings into separate records or adding them on Superwylbur. A disadvantage of Superwylbur is that each physical volume must be input separately if it is to list separately; OCLC provides an item-by-item list from a summary input. The final definition of subfield O to allow LCS libraries to input local data, whether alphanumeric, alpha, or text (e.g., Index), facilitated data entry for complicated law materials and other serials. When subfield L was defined to over-

ride the bibliographic level in the fixed field, libraries with collections of analytics were able to add them. It was not clear to all libraries that the language code in the fixed field generated the LCS language flag (ENG/NONENG), which activated the correct stop list. If libraries did not complete the language fixed field or did not match it to the language of the title, the wrong stop list might apply. Libraries whose OCLC profile identified only one holding library with prefixes or stamps to indicate location had to re-update materials added to OCLC earlier in order to match location or collection with a specific LCS circulation loan period. The full OCLC character set did not translate directly into LCS (e.g., except in the 049 field, square brackets translated as colons and either caused confusion or prevented a search depending on where they appeared in the LCS record). Few libraries had the time or staff to review subject headings to bring them up-to-date; many obsolete headings were left with the expectation that some future machine conversion would make subject access uniform and up-to-date. Although the University of Illinois at Urbana-Champaign implemented AACR2 in November 1979, the other Illinois libraries doing retrospective conversion projects on OCLC for LCS did not have the training and interpretation, nor the time, to incorporate changes for AACR2 in the conversion process. There was also concern that discrepancies between the card catalog and LCS would cause confusion and that AACR2 could not be adopted without providing for recataloging. Again, some global machine conversion would hopefully change all records to AACR2 in the future and match them to an authority file.

When Western Illinois University (Macomb) received a grant from the IBHE for FY 1980–81, they adopted the three-phase plan to begin what was then the largest collection (350,000 monographs and nonperiodical serials) in Illinois to be converted on OCLC. Periodicals were already in non-MARC form on a local database for a regional union list. In planning for retrocon, Western had projected zero cost for the use of the OCLC database, so OCLC's announcement of a database use charge of five cents per record for nonprime time during 1980–81 forced them to halt original cataloging in the last months of FY 1979–80 to update as many records as possible before the new pricing took effect. Millikin University, with 95 percent of the collection converted, was one of the few libraries scheduled for early entry into LCS that almost completed its retrocon projects by the end of the one-year grant period. The other libraries, left with collections still to be converted, changed budget planning to accommodate the new OCLC pricing policy and changed schedules to take advantage of the price difference between prime and nonprime time ($.60 versus $.05) for retrospective conversion projects.

In November 1980 Morris Library, Southern Illinois University at Carbondale, received a grant from IBHE for a full record conversion on OCLC of 492,000 monographic items and twelve thousand serials. SIU-C's sampling of one thousand records from the shelflist indicated a 75.9 percent hit rate with a projected fifteen-records-per-hour conversion rate if they limited editing on the hits to only those fields required by LCS (call number, 049 holding field, and the language code in the fixed field). Carbondale adopted the three passes through the shelflist used by other libraries and scheduled project staff to work in OCLC nonprime time. Carbondale's final hit rate exceeded its projections, reaching 91 percent for monographic conversion.

SIU-C's conversion of serial records also included procedures to adapt the OCLC record for LCS: (1) code the language fixed field to generate the stop list of the title, foreign or English; (2) when appropriate, change the serial-type fixed field to set the loan period for a periodical rather than a serial; (3) edit the 245 field to ensure that essential data appear as the title on LCS (The LCS program reads only subfield A of the 245 field; if needed information is included in following subfields, the delimiters and subfield identifiers must be deleted so that all information appears in subfield A. While this practice affects the search key in LCS for one-word titles, the alternative of adding the necessary information later as an "edition" subfield on LCS would be difficult for a collection as large as SIU-C's.); (4) edit the 260 field to delete square brackets from the place of publication and to include the start date of serial holdings (not the publication start date). As a result of editing for LCS, information that may be needed for an online catalog does not appear on the OCLC archive tape, but the advantage is less review and editing of LCS after the OCLC tape is loaded.

SIU-C's approach to the conversion process benefited from fuller knowledge of the LCS program than the earlier libraries had had in their planning. In addition, Morris Library was able to incorporate AACR2 into its project at an early stage because interpretation and training were then available and the OCLC AACR2 conversion was imminent in December 1980. When Morris Library staff analyzed the impact of available AACR2 records on their project for retrospective conversion of music scores on OCLC, they found 81 percent of the main entries in their sample already changed to AACR2 form.[5]

When Northeastern Illinois University (Chicago) began grant-funded conversion in FY 1981–82, problems with the OCLC/LCS program had either been resolved or could be worked around when planning the project. The completed flip for AACR2 entries in the OCLC database also meant Northeastern's main entries would be current. Entry of complicated holdings in the 049 field was known to be time-consuming and often frustrating so Northeastern joined DePaul and other libraries in deciding not to convert textbooks, curriculum, and other teacher resource materials on OCLC, but to add them directly to LCS via Superwylbur files.

Southern Illinois University School of Medicine Library (Springfield) resumed a spare-time retrospective conversion project when they received an IBHE grant in FY 1981–82. SIU-SM avoided the LCS wrong-call-number problem by leaving only their call number field. All subject headings except National Library of Medicine were deleted, and NLM headings were added if they were not in the record. SIU-SM avoided the problem of adapting their extensive periodical holdings to the complexities of the 049 field by inputting summary statements of noncirculating holdings.

When Eastern Illinois University (Charleston) received an IBHE grant for FY 1982–83, Booth Library staff went back to a retrocon project for monographs and serials that was begun on OCLC in 1978 but temporarily suspended in 1981. All music scores had already been recataloged as part of a separate retrocon project, and another project for recataloging and reclassifying music recordings was under way.[6] The last two institutions to receive IBHE grants for admission to LCS (FY 1983–84) were Elmhurst College (Elmhurst) and Illinois Wesleyan University (Bloomington). Although their grants were for LCS equipment and installation only, both libraries had implemented retrospective conversion projects on OCLC several years earlier in anticipation of joining an online system. Like the initial HECA libraries, both institutions were already committed to OCLC and wanted to do a one-time conversion to have a full bibliographic record for whatever future application might arise.

STATUS OF RETROCON PROJECTS

Although IBHE grants for retrospective conversion have not been awarded since FY 1982–83, retrocon projects are continuing as LCS libraries keep on entering data either via OCLC or directly into LCS. All twenty-three institutions have some materials not yet in LCS, sometimes because the material doesn't circulate but often because the format requires a high level of staff expertise and time. A preliminary survey of materials not yet in LCS shows government documents, periodicals, and microfilms as the largest categories still to be converted. The LCS holdings as of April 1984, however, exceeded 13 million volumes (almost 10 million monographs and more than 3 million serials). A title count shows 7,434,000 monographs and 380,000 serials. During March 1984, the 564 terminals on LCS processed 1,771,669 transactions for bibliographic searching and 794,811 circulation transactions.

The goal of funding the retrospective conversion projects was to promote resource sharing in Illinois. In March 1984, 30,533 interlibrary circulation transactions were recorded, which brings resource-sharing activity past the 600,000 mark since the system was extended to libraries beyond the University of Illinois in July 1980.

REFERENCES

1. Bernard J. Hurley, "LCS: Automated Resource Sharing in Illinois," *Resource Sharing and Library Networks* 1:67–77 (Fall 1981).
2. Diane Foxhill Carothers and William Aguilar, "The Beginnings of LCS at Illinois," *Information Technology and Libraries* 2:393–400 (Dec. 1983).
3. Nancy John, "Preparing for On-line Access: Retrospective Conversion," *Illinois Libraries* 62:618–22 (Sept. 1980).
4. Elaine Rast, "Conversion Strategies for a Machine-Readable Data Base at Northern Illinois University Libraries," *Illinois Libraries* 62:616–18 (Sept. 1980).
5. Richard B. Wursten and James A. Chervinko, "Music Goes On-line: Retrospective Conversion of Card Catalog Records for Music Scores at Morris Library (SIU-C)," *Illinois Libraries* 65:346–48 (May 1983).
6. Phyllis Rearden and John A. Whisler, "Retrospective Conversion at Eastern Illinois University," *Illinois Libraries* 65:343–46 (May 1983). ■■

Retrospective Conversion with REMARC at Johns Hopkins University

Virginia Drake and Mary Paige Smith

On Christmas Eve 1981, the Milton S. Eisenhower Library of the Johns Hopkins University was awarded a $1.2 million grant for the procurement of an integrated online catalog system and the conversion of bibliographic records to machine-readable form. The Online Catalog Committee and the Committee on Retrospective Conversion were formed to recommend to the director the most effective and financially acceptable online system for the library and the most efficient method of retrospective conversion of bibliographic records to be loaded into that system. After months of consultation, demonstration, and solicitation, the recommendations to the committee were to contract with the Carrollton Press to use the REMARC System for retrospective conversion of what is now estimated to be five hundred thousand monographic records; to key serial records directly into the in-house online system when available; and to purchase the Biblio-Techniques Library Information System (BLIS).

Virginia Drake is head of Copy Cataloging, and Mary Paige Smith is retrospective conversion supervisor, both at Milton S. Eisenhower Library, Johns Hopkins University, Baltimore, Maryland.

The decision for REMARC was based on two primary factors: the source and completeness of the database and the facility and efficiency of keying. Since the REMARC database is composed of Library of Congress records, most access points on those records, although not necessarily established under current practice, should appear either as a valid heading or a cross-reference in the LC Authority File. This would allow for global changes either on the local online system or through a vendor. A study to project an expected hit rate using REMARC, in which random shelflist records were searched against the National Union Catalogs, yielded a 73.18 percent hit rate.

The nature of the Eisenhower Library catalogs is such that for many older records, full cataloging information is found only on the main entry card while complete holdings are available only on the shelflist card. The time involved in pulling main entries as well as the subsequent inconvenience to patrons led to the decision that keying local holdings from the shelflist onto an already existing LC bibliographic record was preferable. Finally, the financial benefits of using Apple microcomputers rather than OCLC or RLIN terminals and of employing key operators rather than trained catalogers to key simple searches and tag local data confirmed that REMARC was indeed a good choice.

THE REMARC SYSTEM

The REMARC database, when complete, will contain the MARC file of bibliographic records, plus all Library of Congress roman-alphabet cataloging records converted to machine-readable form by the Carrollton Press, for a total of nearly seven million records by the end of 1984. Keying of the database, which is currently in progress in Glasgow, Scotland, is done from photocopies of Library of Congress shelflist cards arranged in title order and sorted by language. The records pass through LC's format recognition program and are then edited by the Carrollton Press. As of this writing, all English-language titles have been keyed as well as two-thirds of the non-English titles, although not all have been edited and made available.

It should be noted that these records are partial, lacking some fixed field elements, the b and c subfields of the 245 field (bibliographic title), series notes (4xx fields), all notes except the "title romanized" note, and, in some instances, the collation.

THE PROCEDURE

Before keying, each drawer of the shelflist is sorted by the REMARC project supervisor or one of the full-time key operators. Cards that should not be keyed (OCLC, RLIN, non-roman, serials, etc.) are marked, and local holdings information to be added to the REMARC record is highlighted. Since analytics for monographic series classified as a whole are not kept in the shelflist, the face card is pulled from the shelflist and the series title checked in the catalog. If analyzed, corresponding main entries for all analytics are pulled from the author-title catalog. This is done by student assistants or by full-time key operators during "off-terminal" time.

After sorting, key operators are assigned segments of approximately four inches of shelflist cards and enter search requests onto floppy disks on Apple microcomputers. The search request consists either of the LCCN (if known) and the first two words of the title or, if the LCCN is not available, of the first twenty-three characters of the title and three characters from the place and date of publication. To the search result the key operator adds local holdings (call number, variant locations and copy information, replacement notes), series notes (4xx), bound-with notes, and uniform titles added by catalogers under AACR2.

Completed disks are sent to the Carrollton Press where the search keys are matched against the REMARC database. Hits are returned to the library in tape form. Printouts are returned to the library for multiple hits (search key retrieves more than one record, e.g., Title-Collected works), duplicates, and problems. These must be rechecked individually against the shelflist and returned to the Carrollton Press with the solution.

STAFFING

It was decided that, in order to take full advantage of the three Apple microcomputers loaned to the Eisenhower Library for the project by Carrollton Press, support staff should be hired to work in shifts from 8 a.m. to 8 p.m., Monday through Friday. The project began on February 14, 1983, with three full-time and six part-time data entry operators (DEO) and a full-time supervisor (formerly a copy cataloger in the department). The part-time DEOs worked four hours a day, keying for two hours at a time with a break between sessions. The full-time DEOs keyed for four hours a day, in two-hour shifts; their remaining time was spent doing card pulling and filing related to the project.

The biggest problem in the early days of the project was keeping up with who had keyed what and who had revised whom. There were nine keyers and one full-time reviser (several Cataloging Department staff members also did part-time revising), which resulted in a backlog of revising. The problem was made less serious by the early decision not to fill disks with as many keys as they would hold. Each shelflist section was keyed on a separate disk, so that each disk contained approximately 250 to 350 search keys.

Another problem stemming from the extended hours of the project was that of multiple supervisors. There was a different supervisor (from the regular Cataloging Department staff) almost every night. A form was developed to enable evening supervisors to communicate questions, problems, or information to the regular daytime supervisor.

Once the DEOs were on their way to being fully trained and the system was running fairly smoothly, it became obvious that the method of preparing shelflist drawers for input was not satisfactory. In order to expedite keying, each shelflist drawer was first examined by a senior member of the Cataloging Department. This process enabled DEOs to spot the data that needed to be keyed from each card, and resulted in a keying speed after training of 80 to 120 searches per hour. However, since regular Cataloging Department staff were doing the sorting of shelflist drawers, they could not give the amount of time per week necessary to keep up with the DEOs. At one point several months into the project, only 50 percent of scheduled computer time was being used, as DEOs keyed searches twice as fast as drawers could be prepared for them. The solution seemed to be to have the DEOs, most of whom were fully trained at this point, assume some of the responsibility for preparing shelflist drawers. This solution has proved successful; full-time DEOs now prepare at least one shelflist drawer per day for input.

As soon as the DEOs had enough work to keep the computers busy twelve hours a day, the card pulling and filing backlogs grew by leaps and bounds. A full-time student assistant was hired for the summer; by fall it was clear that at least ninety hours of student help per week was needed to keep up with the work load. Even with a team of six students, there are still substantial backlogs of series cards to be checked and analytics to be pulled.

An ongoing problem has been staff turnover, especially in the part-time ranks. In addition to the time that must be invested in searching for a new employee, productivity is lowered both in the interim between the loss of an old employee and the hiring of a new one, as well as during the training of a new staff member. In order to assist in the training of new employees and the supervision of the evening shift, a full-time lead DEO position was created. The incumbent, one of the original DEOs, performs a variety of tasks besides keying, such as making backup copies of disks, preparing mailings of disks to Carrollton Press, and training both new permanent staff and student assistants in the various phases of their jobs.

DOCUMENTATION

Carrollton Press provided basic instructions for the Appleskip program, which is used to key searches. The basic local documentation, which incorporates the system documentation, was written by the assistant head of the Cataloging Department before the project was begun. Although it has undergone minor modifications and expansions, this document has served the project well. Other logs and statistics developed for the project are described below.

Permanent disk log

This log is used to record the number of each disk, the name of the data entry operator who keyed it, the date it was begun and finished, the beginning and ending call numbers on the disk, and the date the disk was sent to Carrollton Press. The disk log furnishes a permanent record of the contents of each disk.

Box log

Used by each DEO to record the number of the disk being keyed, the box log provides a record of the number of search keys on each disk at the end of each keying session, as well as the beginning and ending call numbers keyed at each session. This log keeps track of each shelflist section being keyed and records the keying speed of each typist.

Supervisor's log

This log was used in the early days of the project to record the activities of the unit during both the day and evening shifts. As soon as the staff established a stable working pattern and many of the initial questions about the project had been answered, the use of this log was discontinued.

Apple problem log

This is the supervisor's record of hardware and software problems. The main problem experienced so far has been the failure of the enhancement that allows keying of upper- and lowercase characters on the Apple II+. Two of the three original Apple II+ computers have been replaced by Apple IIe's, which do not require this enhancement.

Error sheet

This log is used by the reviser of a new DEO to record the errors in keying that need to be corrected. The form provides space for the record number of an errored key, as well as an explanation of the nature of the error and instructions for correcting it.

Monthly statistics

This form is updated monthly to reflect new totals—for the current month, fiscal

year, and total project—in the following areas: number of disks transmitted to Carrollton Press, number of search keys on the disks sent, and the number of shelflist drawers represented by these search keys. Also recorded are the number of search keys read by Carrollton Press in their monthly match run and the number of resulting matches.

CLEANING UP

Actual keying of REMARC searches is expected to be completed in September 1984, one and a half years after the project began. As soon as the REMARC database is complete, Carrollton Press will make a final match of all remaining nonhits, sending the library tapes of the hits and a list of nonhits in call number order. Nonhits may be directly keyed into the BLIS system, concurrent with the "blue pass" (an update of those bibliographic records to which changes have been made since REMARC keying took place, and so designated by a blue card in the shelflist). The alternative to local keying of nonhits is to catalog them on RLIN. No decision has yet been made on this issue.

As of this writing, only a small percentage of the unique hits have been checked against the shelflist to verify a match. This verification was done using printouts of hits, including local holdings, which were furnished by Carrollton Press as requested in the initial contract. No REMARC tapes have yet been loaded onto BLIS, although one of the tapes will form part of the acceptance testing for the local online system. It is a foregone conclusion that every hit cannot be checked before the online catalog becomes available to patrons. It is hoped that checking done for the blue pass and for resolution of multiple hits will provide the assurance that the REMARC matching process is indeed reliable.

CONCLUSION

Since the work is yet to be completed and the final product remains to be seen, it is impossible to analyze the successes and failures of the REMARC project at Eisenhower. It is not too early, however, to offer advice and to recount some lessons learned.

Above all, experimentation should be done, preliminary debugging completed, and detailed procedures written before staff is hired. Catalogers at Eisenhower spent many frenzied hours dreaming up eventualities and testing, writing, and rewriting procedures. The resultant ease and speed with which the DEOs were trained, however, made the struggle well worthwhile.

Secondly, supervisors should be encouraged to note problems or discrepancies in procedures and be sure that everyone involved learns of the solutions. It is exceedingly easy in a project of this magnitude and with several supervisors to have varied solutions to the same problem. As trivial as the issue may seem at the time, it could later affect access to data in an online catalog.

Lastly, it is important to realize that very few retrospective conversion records will be complete down to the last detail. It is probably better to accept this fact and try to strike a balance somewhere between complete, flawless data and useless online garbage. At the outset of the project, it is advisable to include as much data as reasonable with the expectation that some elements will be eliminated later to increase productivity and lower costs. ■■

Authority Control in the Retrospective Conversion Process

Dan Miller

Retrospective conversion of a library's records to machine-readable form in support of automated circulation control, or for an online or COM catalog, is either in process or on the agenda of many libraries. Conversions are typically undertaken because of the improved service and long-range cost savings derived from working in an automated environment. Much has been written on retrospective conversion options with survey articles reflecting the technological innovations over time. Butler

Dan Miller is manager of sales and service, Technical Services Division, Blackwell North America, Lake Oswego, Oregon.

et al.[1] reviewed the offerings and procedures when retrospective conversions were first gaining momentum. More recently, Epstein,[2] in a series of articles, brought the subject up-to-date with an exploration of the vendors and methods associated with retrospective conversion and the nuances associated with converting copy-specific information. Secondary concerns relating to duplicate record elimination and holdings consolidation in a single source file at a major academic library are explained by Caplan.[3] Her article details Harvard University's method for consolidating a MARC transaction file, where each record represents a volume and local information resides in various fields, into a database where each record represents a title and local holdings data are consolidated into a single field. McPherson et al.[4] describe the University of California's method for merging records from various sources for the production of an online union catalog.

WHY APPLY AUTHORITY CONTROL?

One aspect of the retrospective conversion process that has not received much attention in the literature is authority control. Most large source files used in the retrospective conversion process contain records in which the heading fields contain data reflecting the rules and practices in effect at the time the records were converted, or at the time the cataloging for the title was originally done. With regard to name and uniform title access points, the implementation of AACR2 has tended to compromise the uniformity of the data, as some records were created prior to implementation of AACR2 and some were created after implementation. For subject access fields, our perpetual redefinition of the world around us along with changes in cataloging rules has resulted in source files in which works for given subjects are likely to be found under various (sometimes archaic) headings. These inconsistencies, coupled with other types of heading errors, impede the colocation of entries in a traditional COM catalog and inhibit access to records in an online catalog. A system for correcting headings in converted records via error-correcting algorithms (like those described by Brown[5] with regard to OCLC's AACR2 edit), or via authority file matching, is therefore required.

B/NA'S AUTHORITY CONTROL SYSTEM

The Blackwell North America (B/NA) authority control system has been in operation for more than thirteen years. While subject authority control in support of B/NA-produced COM catalogs was the principal focus of the system for many years, recent activities reflect equal emphasis on both name and subject authority control in support of both COM and online catalogs. The authority control system is a batch mode, offline system. It employs a file-matching approach involving libraries' converted bibliographic files and Library of Congress name and subject authority files. In addition to authority data received from LC in machine-readable form, the subject authority file includes more current records keyed from LC's printed lists of subject headings. Also, the system extracts very new headings from cataloging processed through the system. These headings perform a standardizing function temporarily until replaced by a full authority record provided by LC.

Briefly stated, the system does the following:

- Updates library headings in the MARC format to LC's most recent practices.
- Standardizes the forms of headings for filing purposes and for improved online access.
- Corrects MARC tags and subfield codes based on data content.
- Provides deblinded cross-references for use in COM catalog production or online systems.

The first three capabilities relate to the findings of O'Neill and Aluri, later expanded upon by O'Neill and Vizine-Goetz, concerning the majority of errors found in subject-heading fields of 33,455 OCLC sample records.[6] The authors identify four major categories of errors:

1. Inconsistency in spacing, punctuation, and capitalization.
2. Typographical and minor spelling errors.
3. Invalid form of heading.
4. Incorrect MARC tag or subfield code.

Before explaining in detail how the authority system addresses these errors, I would like to briefly review the structure of authority records (see figure 1). Each record contains a field with the authorized data and, frequently, other fields, such as *see from* fields, containing old or unauthorized forms of the same concept or name. Library headings are matched against *see from* data to determine if upgrading is required and against the authorized form to validate the heading. *See also from* data, which will not be examined here in depth, provides information on related headings. Understanding how correction of the aforementioned errors is accomplished requires an understanding of the difference between the normalized form of a heading and the catalog form of a heading.

NORMALIZED VERSUS CATALOG FORM OF HEADING

The normalized form of a heading is a computer-edited form (see figure 2), where all alphabetic characters are set in uppercase, MARC tags and subfield codes removed, all punctuation and special and diacritical characters deleted, and spacing between words regularized. The catalog form of the heading is the form that we are accustomed to seeing in a full MARC display. It includes full punctuation and capitalization, all special and diacritical characters, as well as MARC tags and subfield codes.

When the authority system passes headings from a library catalog against the authority file, the headings are submitted to a matching test at two levels. First, matches are identified based on comparison of the normalized form of the library heading with the normalized form of the authority record heading. The same headings are then matched using the catalog form of the headings. Any difference in headings at the catalog form match phase is attributed to an error in the library heading, and the system replaces it with the form from the authority file.

In figure 2, the normalized forms of the headings from the authority file and the bibliographic file match, but the catalog forms do not match, as the library heading is missing diacritical marks and the spacing in subfield d is in error. The system replaces the library's heading with the correct form from the authority file.

ERROR CORRECTION

The first of O'Neill's categories of errors is inconsistencies in spacing, punctuation, and capitalization. As a result of the two-level match between the normalized form and the catalog form of headings, such inconsistencies are eliminated.

O'Neill's second category is typographical and minor spelling errors. B/NA's editorial staff has found that this category constitutes the largest portion of exceptions or nonmatches through the system. These exceptions are easily identified and remedied by the staff of editors because, as O'Neill points out, they are usually simple errors of omission, addition, substitution, or transposition of characters. Once these errors are corrected by the editors, the headings

150 AUTHORIZED DATA	150 CLARINET
450 SEE FROM DATA	450 CLARIONET
450 SEE FROM DATA	450 PRIMER CLARINET
550 SEE ALSO FROM DATA	550 WOODWIND INSTRUMENTS

Fig. 1.

Library Heading

600$aDvorak, Anton,$d1841- 1904

Normalized Form

DVORAK ANTON 1841 1904 ◄——— AUTHORITY FILE

DVORAK ANTON 1841 1904 ◄——— LIBRARY FILE

Catalog Form

100$aDvořák, Anton,$d1841-1904 ◄——— AUTHORITY FILE

600$aDvorak, Anton,$d1841- 1904 ◄——— LIBRARY FILE

Fig. 2.

are again passed against the authority file for upgrading to the proper form of the heading or to validate the heading as the correct form.

The third category noted by O'Neill is invalid form of the heading, including incorrect use of qualifiers, inversion errors, and abbreviations (such as "U.S." where "United States" is required). Incorrect use of a qualifier, such as in the heading "Phylogeny (Zoology)," is automatically remedied when this form exists as a *see from* field in an authority record. The system supplants the erroneous form on the library's file with the correct form "Phylogeny."

An inversion error, such as in the heading "Inorganic Chemistry" is automatically changed to "Chemistry, Inorganic" when the erroneous form exists as a *see from* field in an authority record. However, many inversion errors and incorrectly used qualifiers and most abbreviations (O'Neill's third category of form error) do not exist as *see from* data in LC authority records. B/NA maintains a machine-readable file of thirty-five hundred commonly used erroneous abbreviations, inversion errors, and incorrectly used qualifiers that LC has chosen not to include in its authority file. They have been added to B/NA's authority file in a manner that allows the system to supply the correct form of the heading to the bibliographic record without providing a *see* reference from the erroneous forms. Thus, the headings are corrected and at the same time the integrity of LC's cross-reference structure for a given heading is maintained.

O'Neill's fourth category is incorrect MARC tags or subfield codes. Authority file headings contain tags that are similar to bibliographic file headings in that the last two digits of the tag indicate the type of heading. For example, corporate name fields 110, 410, 610, 710, and 810 in bibliographic records all end with 10, while the authorized form of a corporate name in an authority record is tagged 110 (see figure 3). When a data match occurs, a routine compares the last two digits of the MARC tags from the authority file heading and from the library heading and corrects the tag in the library heading field, if necessary, to match the last two digits on the authority heading tag. The headings in figure

Type of Heading	MARC Tag Suffix
Personal Name	00
Corporate Name	10
Conference Name	11
Uniform Title Subject	30
Topical Subject	50
Geographic Name Subject	51

```
100$aDvořák, Anton,$d1848-1904 ◄── AUTHORITY FILE

610$aDvořák, Anton,$d1848-1904 ◄── LIBRARY FILE
```

Fig. 3.

3 match, but the last two digits of the tags do not. The system changes the tag on the library heading to 600.

As matching between library headings and authority file headings involves the two-phased match of the normalized form (in which subfield codes are absent) and the catalog form (where the subfield codes are present), all subfield codes from the authority file heading are imposed on the library heading. This is not unlike the logic that corrects spacing and diacritical errors, as the subfield codes are simply treated as data. In figure 4 the normalized forms of the headings from the authority file and the bibliographic file match, but the catalog forms do not, as subfield code d is miscoded as b. The system replaces the library's heading with the correct form from the authority file, including the correct subfield code. This feature of the system is particularly useful for correcting errors in subject subdivisions where subfields that should be coded y (period) and z (place) are often miscoded as x (general).

OTHER EDITS

A condition not mentioned by O'Neill but addressed by the system is where an existing heading can be changed to more than one heading depending upon the context. For example, "Colonialism" can be changed to "Colonies," "Imperialism," or "World politics." In this case, the records are displayed for review by editors who choose the appropriate heading or headings for assignment to the record.

If a full heading (including subheadings)

Library Heading

600$aDvorak, Anton,$b1841-1904

Normalized Form

DVORAK ANTON 1841 1904 ← AUTHORITY FILE

DVORAK ANTON 1841 1904 ← LIBRARY FILE

Catalog Form

100$aDvořák, Anton,$d1841-1904 ← AUTHORITY FILE

600$aDvorak, Anton $b1841-1904 ← LIBRARY FILE

Fig. 4.

is not found on the authority file during the match process, but the main heading and each subdivision exist independently on other authority records, the system views the full heading as permissible. After editorial review, it is added to the authority file as a new valid combination. This "parts match" also operates where the main heading or subheadings exist as *see from* fields in authority records, in which case the heading or subheading is changed to its authorized form before the full heading is deemed permissible. As all matching involves a two-phased match between normalized and catalog forms of the heading, parts matches also correct errors in punctuation, spacing, capitalization, and special or diacritical characters.

Conversion of a library's authority file to machine-readable form is also addressed by the system as it generates a tape copy of all authority records matched or created during the authority control edit. Duplicate records are eliminated from this file, and all *see also from* references are deblinded to correspond to headings on the library's bibliographic file. The provision of this file, in a format compatible with *Authorities: A MARC Format*, along with a copy of the library's edited bibliographic file, completes the library's retrospective conversion process.

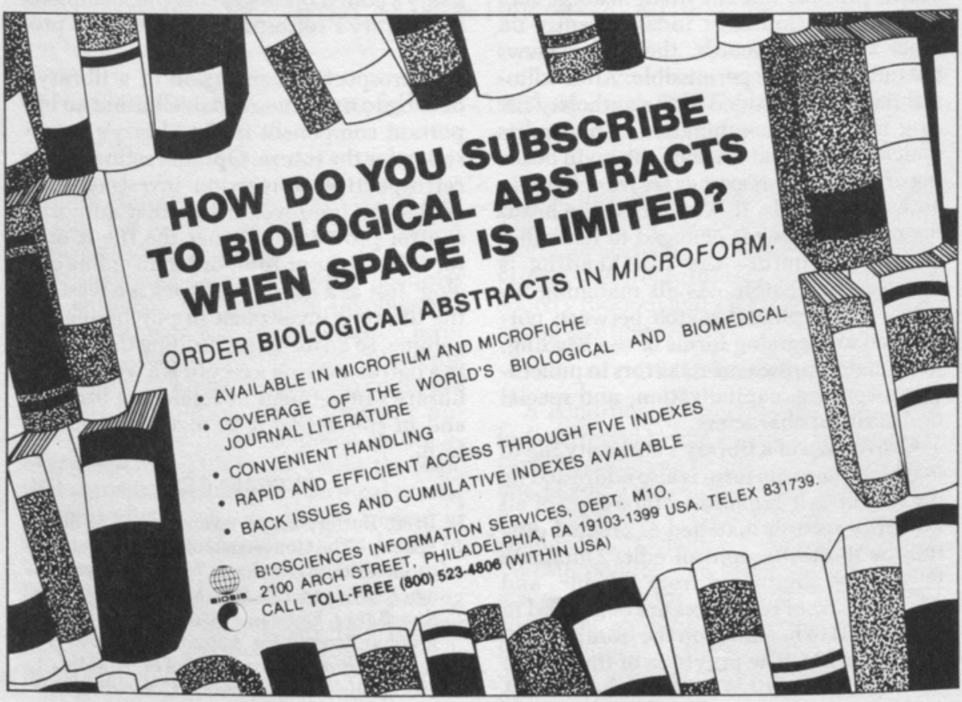
Retrospective conversion of a library's catalog to machine-readable form is an important component in the library's preparation for the future. Optimal return on the retrospective conversion investment includes the improved access that authority control provides, whether the file is used for COM fiche or film, or for an online catalog. Just as a misshelved book is a waste of the library's investment in purchasing the volume, so an incorrect heading that results in a patron missing a record is a waste of the library's investment in cataloging the item and in converting it to machine-readable form.

REFERENCES

1. Brett Butler, Brian Aveney, and William Scholz, "The Conversion of Manual Catalogs to Collection Databases," *Library Technology Reports* 14:109–206 (Mar.–Apr. 1978).
2. Sue Baerg Epstein, "Converting Bibliographic Records for Automation: Some Options," *Library Journal* 108:474–76 (Mar. 1, 1983); Sue Baerg Epstein, "Converting Rec-

ords for Automation at the Copy Level," *Library Journal* 108:642–43 (Apr. 1, 1983).

3. Priscilla Caplan, "Retrospective Duplicate Resolution for the Harvard Distributable Union Catalog," *Information Technology and Libraries* 1:142–43 (June 1982).
4. Dorothy S. McPherson, Karen E. Coyle, and Teresa L. Montgomery, "Building a Merged Bibliographic Database: The University of California Experience," *Information Technology and Libraries* 1:371–80 (Dec. 1982).
5. Georgia L. Brown, "AACR2: OCLC's Implementation and Database Conversion," *Journal of Library Automation* 14:161–73 (Sept. 1981).
6. Edward T. O'Neill and Rao Aluri, *A Method for Correcting Typographical Errors in Subject Headings in OCLC Records*, Report no.: OCLC/OPR/RR-80/3. (Columbus, Ohio: OCLC, 1980); Edward T. O'Neill and Diane Vizine-Goetz, "Computer Generation of a Subject Authority File," *Proceedings of the ASIS Annual Meeting* 19:220–23 (1982). These two articles are hereafter cited as O'Neill.



Communications

Normalization: A Method for Structured File Design

Jerry V. Caswell

INTRODUCTION

Library automation requires an extraordinary investment of time and money in the development and maintenance of databases. Irrespective of whether a library sets up a stand-alone system such as circulation, a group of more or less related subsystems, or a totally integrated system, it can be certain of one thing: over time needs will change, new applications will have to be developed, and modifications will have to be made to the file structure of the databases. Of principal concern to the designers of a library's databases should be finding a design structure that will allow modifications to be made with a minimum of additional time and cost.

THE MASTER RECORD APPROACH

In the past, when files were computerized, the structure of manual files provided the model for computerized ones because it was felt that file structure should be determined by the needs of the application. In manual files a paper form often acted as the master record for a particular activity. On that form all relevant data elements were recorded, some data elements multiple times. Eventually the form was filled and another had to be started.

The same principle was applied to the construction of computerized files. The programmer created a master record bristling with every applicable data element. Where certain data elements had to be repeated, the programmer estimated the maximum frequency of the repeated data elements and built that into the files. Retiring a master record in a computerized file was more complicated than in a manual file, however. Most computerized files were sequential in nature and processed in batch mode, which means that one could not replace a single record in the middle of a file without processing the file from beginning to end. Although the later introduction of direct access files mitigated this problem, the fact remained that lumping all data elements together was an awkward way of building files and that predicting the number of occurrences of repeating elements was an uncertain art.

There were other problems with such files as well. The insertion, modification, and deletion of records caused certain anomalies to appear. Programmers often had to circumvent these anomalies by making their programs nonmodular and idiosyncratic. In order to make these issues clearer, let me illustrate them with an example from a library application.

Assume that a reserve collection is being computerized and that a master reserve record will contain most of the needed data elements. The data elements describe at least three different entities: the bibliographic item, the course for which it is on reserve, and the instructor who teaches the course. In order to keep the illustration simple, I shall identify only a few of the data elements applicable to each entity.

The data elements pertaining to the bibliographic item are the call number, the author, the title of the work, and the semester during which it is used on reserve. The course information includes the course number and the name of the course. Because a bibliographic item may be on reserve for more than one course during a given semester, data elements have been

Jerry V. Caswell is head of the Systems Dept., Duke University, and a doctoral student at the University of Wisconsin–Madison Library School.

provided for two course names and numbers. Let us assume that each course is taught by one instructor. Therefore, there is an instructor's name and phone number to correspond to each course number and name. The file structure of the Reserve file would look like this:

RESERVE (Call number, Author, Title, Semester,
Course number 1, Course name 1, Instr. name 1, Instr. phone 1,
Course number 2, Course name 2, Instr. name 2, Instr. phone 2)

The drawback to repeating data elements might be that 10 percent of the bibliographic items are on reserve for more than one course. Nevertheless, space for the second course and instructor data elements had to be committed at the time the records were created. Thus, 90 percent of the records contain unused and wasted space.

THE PROBLEM OF ANOMALIES

The design also presents other problems. If a new course is offered, the file structure is unable to record that fact until at least one bibliographic item has been placed on reserve for it. The same is true for new instructor data. Because both course and instructor data are dependent upon the existence of a bibliographic entry, the file structure does not allow independent recording of course and instructor information. This is called the insertion anomaly.[1]

Conversely, under the file structure outlined above the removal of the last bibliographic item for a given course would delete all references to that course and its instructor. This is called the deletion anomaly.[2]

Modifying any of the data elements creates some unique problems. For example, updating an instructor's phone number would require searching through the entire Reserve file, looking at all the repeating instructors' names, and updating the phone number wherever the appropriate instructor's name appeared. The same procedure would apply to updating the data elements for either the course or the bibliographic item. There might be several occurrences of each entity in the file, and each occurrence would have to be updated or the data would be out of synchronization. Because it is easy for some occurrences of a data element to get out of synchronization with other occurrences, this problem is called the update anomaly.[3]

The consequences of the "master record" approach to file design are that data elements are bound in inflexible combinations with other data elements, that new needs require completely new file design, often with the replication of existing data, and that files cannot be shared among different, but related, applications. The practical consequences of this are both economic (increased programming costs) and temporal (later implementation dates).

NORMALIZATION

One method of designing databases to avoid the problems of the "master record" approach is to use *normalization*, a process for designing file structures so that they are independent of a particular application. Such independence makes files more modular and hence less subject to change when new applications are developed. This means that existing files may be shared among applications, that less programming and maintenance time will be required, and that programming projects will be completed sooner.

Normalization is a method of constructing files that eliminates the problems of repeating data elements and the insertion, deletion, and update anomalies. The principle behind normalization is that file design should be based upon the relationships between the data elements rather than upon the application that the files serve. Observing this principle in the design of file structures makes them independent of any specific application. The process of normalizing a file involves defining the relationships between the data elements and "decomposing" the data elements into a series of files based upon those relationships. There are at least three steps to this process, each of which results in a further refinement of what is called a *normal form*. Let us use the Reserve file described above to illustrate how decomposition may be used.

PRIMARY KEYS

Before proceeding to decomposition, however, it will be necessary to explain the concept of a key, or, more specifically, of a primary key. In the normalization process, a key is that data element or group of elements that uniquely identifies a particular entity. It distinguishes that entity from all others of its type. If we apply this concept to the Reserve file, we can say that the call number and semester taken together are the key for each record because together they uniquely identify any record in the Reserve file. According to the conventions of normalization, data elements used as keys may be starred in file descriptions:

RESERVE (*Call number, Author, Title, *Semester,
Course number 1, Course name, 1, Instr. name 1, Instr. phone 1,
Course number 2, Course name 2, Instr. name 2, Instr. phone 2)

DECOMPOSITION AND THE THREE NORMAL FORMS

The process of decomposition starts with the principle that the data elements in each record should be atomic, that is, there may be no repeating elements. Accordingly, repeating data elements should be removed from the master record and placed in a separate file with an appropriate data element as key. Once all repeating elements have been removed, each of the data elements in the record is said to be atomic and the record is in *first normal form* (1NF).[4]

In order to bring the reserve records into conformity with first normal form, the course number and name and the instructor's name and phone number should be removed and placed in a file called Course. Since there is a relationship between the bibliographic items and the courses they serve, the course number is retained in the Reserve file as part of the key. The composite key of the Reserve file now defines a reserve record as a bibliographic item (represented by a unique call number) on reserve for a certain course (course number) during a given semester. The Course file indicates that a certain course has a certain title and is taught by an instructor with a certain phone number. Each data element is now atomic and all the files are "flat."

RESERVE (*Call number, Author, Title, *Semester, *Course number)
COURSE (*Number, Name, Instructor's name, Instructor's phone)

Removing repeating data elements from the master file simplifies the file structure and partially solves the problems of insertion, deletion, and update. The course data are grouped together in one file, and the bibliographic data in another. However, the process is not yet complete because the principles of normalization predicate that the data elements in a file should be fully dependent upon the key.[5] In the Reserve file the bibliographic data elements (author and title) are dependent upon part of the key (call number) and not upon the semester or course number, which also constitute the key. Hence the nonkey elements should be moved into a separate file with their own key. This results in a state called *second normal form* (2NF).

RESERVE (*Call number, *Semester, *Course number)
COURSE (*Number, Name, Instructor's name, Instructor's phone)
ITEM (*Call number, Author, Title)

The principles of normalization also specify that all nonkey elements in a file be dependent only upon the key and not upon some intermediary element.[6] Note that in the Course file the phone number of the instructor is dependent upon the instructor's name, which in turn is dependent upon the course. This kind of relationship is called a *transitive dependency*. This relationship should be corrected by moving the instructor's phone number into a separate file with its own key. Once that has been done, the file structure is said to be in *third normal* form (3NF).

RESERVE (*Call number, *Semester, *Course number)
COURSE (*Number, Name, Instructor's name)
ITEM (*Call number, Author, Title)
INSTRUCTOR (*Name, Phone)

It is possible to become even more rigorous through the application of *Boyce-Codd* *normal form* (BCNF) and *fourth and fifth normal forms* (4NF and 5NF).[7] One could ask, for example, whether in the file Item the title is dependent upon the author rather than upon the call number. Alternatively, one could propose that the call number is a function of the title or author/title instead of vice versa. However, for most purposes, reducing files to third normal form is sufficient to achieve the strength that comes with modularity. With the structure outlined above, it is possible to carry out the standard insertion, deletion, and update functions without causing the anomalies that would have resulted from the original design.

THE ELIMINATION OF ANOMALIES

Let us look at the anomalies that have been eliminated. Remember that in the original design updating an instructor's phone number would have required searching through the entire Reserve file, looking at all the instructors' names, and updating the corresponding phone number wherever the appropriate instructor's name appeared. With the normalized set of files, updating a phone number requires finding only a single record in the Instructor file and modifying it. The same applies to updating elements for the bibliographic item or the course information. Only one record for each entity exists in the fully normalized file set, which simplifies updating a record.

Likewise, the normalized file set does away with the insertion anomaly. If a new course is being offered, the original file structure would have been unable to take note of it until at least one bibliographic item had been placed on reserve for it. In the normalized file set, it would be possible to record a new course or any new entity, including an instructor or a bibliographic item, without creating a reserve record for it. A reserve record could be created at the time it was needed to indicate that a specific item had been placed on reserve.

In like manner the normalized file set eliminates the deletion anomaly. Under the original file structure, the removal of the last bibliographic item for a given course would delete all references to that course and its instructor. In the normalized file set, a reserve record may be deleted without affecting the status of the bibliographic item, instructor, or course entities. This means that information for those entities may be kept in the database and used only when needed to construct a reserve record. This would provide a significant decrease in data entry time.

CONCLUSION

The principles of normalization were annunciated in the early 1970s by E. F. Codd.[8] Codd used them as the mathematical foundation of relational database management systems. However, it has become clear that the principles of normalization may be applied to any computerized file structure with beneficial results.[9]

With normalized files, changes to the file structure will not require substantial redesign because unrelated data elements are in separate files rather than tied together in accordance with the old "master record" concept. Second, there will be fewer data elements throughout the entire file set because repeating data elements have been eliminated. Third, new applications will need a minimal number of new data elements because existing data elements and files may be shared. Fourth, because data elements are more flexible, new programs are easier to write and existing ones are easier to modify.[10]

These benefits are not achieved without cost, however. Most libraries with applications already in place will find that a massive redesign effort will be required in order to obtain the benefits of normalization. For static applications it may not be worth the effort required to redesign the database structure. However, for systems that are under development or being altered for one reason or another, normalization offers distinct advantages over traditional file structures, especially when used in coordination with modern techniques such as structured analysis and structured programming.[11]

REFERENCES

1. C. J. Date, *An Introduction to Database Systems,* 3d ed. (Reading, Mass.: Addison-Wesley, 1981), p.69.
2. Ibid., p.70.
3. Ibid., p.70.
4. Ibid., 243–45.

5. Ibid., p.246.
6. Ibid., p.248–49.
7. Ibid., p.249–63.
8. E. F. Codd, "Further Normalization of the Data Base Relational Model," in *Data Base Systems*, ed. R. Rustin (Englewood Cliffs, N.J.: Prentice-Hall, 1972.)
9. Lon Schoor, "Data Normalization" (University of Wisconsin–Madison, Administrative Data Processing, December 10, 1982, Photocopy), p.1.
10. William R. Durrell, "Cohesion: A Means to Data Base Flexibility," *ComputerWorld* 17:ID29–ID32. (October 10, 1983).
11. Schoor, "Data Normalization," p.1. ■■

Database Management for Interlibrary Loan

Matthew J. Kane

A database management system can enable an interlibrary loan department to provide more efficient service by eliminating some routine tasks and by making it easier to organize procedures. This has been the case at the Altoona Area Public Library in Altoona, Pennsylvania. There the staff use an Apple III microcomputer, a 5 MB hard disk drive, and an Apple Letter Quality Printer together with PFS: File and PFS: Report software to handle interlibrary loan transactions.

PLANNING STAGES

Through an LSCA Title III grant from the State Library of Pennsylvania, the public library and the Altoona Area High School Library worked together to provide microcomputers for use by their clientele and their staffs. In preparation for these activities, at least ten library staff members took computer literacy courses.

One of the primary areas of service that the public library focused on was interlibrary loan. Prior to the installation of the microcomputer system, steps were being taken to help make interlibrary loan procedures more efficient. As a district library center, the library provides direct service to thirteen public libraries in its three county district, as well as to other district libraries in Pennsylvania and to any other libraries requesting interlibrary loan service. The Altoona Library is a member of OCLC through the Pittsburgh Regional Library Center.

During the past reporting year (1982–83), the library filled 4,423 interlibrary loan requests. Approximately half of these were for libraries within its district. In order to speed up response time (i.e., the time it takes either to send the item requested or to report on its status) to libraries in the district, Altoona changed its request form so that the response slip had most of the information about the item filled out before the request was received. In this way, the staff merely had to check off the current status of the item and return the response to the requesting library if the item was not immediately available. This cut the response time from several weeks to a maximum of one week, which satisfied the needs of district librarians. However, a survey indicated that there was some dissatisfaction with the amount of time that it took to receive items that were either referred or on reserve.

The anticipated advantages of using a microcomputer for interlibrary loan were that

1. Reserve lists would be printed out in shelflist order, thus eliminating much filing. Reserve lists are necessary because the library's photographic circulation system does not allow for a method of flagging reserves as they are returned to the library.
2. The logging in of items when they are sent out of the library and returned would be done using a printer, rather than doing it by hand in a written log.
3. Statistics reporting, which took up to four hours a month, would be done more quickly.

The anticipated disadvantages of using a microcomputer were

1. A lot of time would be spent inputting information, particularly at the beginning of the project when a database was being built.
2. The possibility of mechanical failure was a concern, and time must be spent in copying information onto floppy disks for protection.

Matthew J. Kane is the library consultant, Altoona Area Public Library, Altoona, Pennsylvania.

IMPLEMENTATION

With these objectives and precautions in mind, the staff began in June 1983 to input all of its active interlibrary loan transactions onto the Apple III microcomputer and hard disk drive. A college student on the Title III Summer Youth Employment Program was able to spend up to twenty hours per week during the summer working on the project. It was important to have the added help at that point, because the interlibrary loan librarian, who also works part-time at the reference desk, would not have been able to do all of the inputting.

The fields that were constructed on the PFS: File database resemble an interlibrary loan request form (see figure 1). In addition to bibliographic information and patron and library information, a field entitled "PLACE" is included. For the purpose of in-house statistics and statistics for the State Library of Pennsylvania, requests are divided into three categories: requests from within the three-county district, requests from other District Library Centers in Pennsylvania, and requests from all other libraries. A different one-letter symbol is given for each category so that, at the end of the month, transactions can be divided into these categories before the statistics are printed out.

By September 1, 1983, all of the active transactions had been entered into the microcomputer's database. The more than 1,200 transactions had been given the status of "Reserved," "Referred," "Sent," or "Returned." There were 636 items on the reserve. With our first reserve list printout, 169 books were found by searching the shelves. Using the list was found to be much easier for searching than using the request forms. After those items were sent out, an additional 146 reserves were determined to be too old to continue searching, and so they were able to be quickly eliminated from the database by merely instructing the computer to erase any reserves placed before a given date.

Another step that was taken in order to shrink our large list of reserves was to request items from the other libraries in our district whenever possible. The Altoona library keeps a union catalog of all the books held by the other public libraries in the district. Before, this catalog was only referred to if a book was not in Altoona's own card catalog. Now, if a book is listed in the Altoona card catalog, but is not on the shelf, then the union catalog is used, and if a card is found, the request is referred to the smallest library that owns the book. In this way, the burden is spread evenly among the libraries, and requests are filled more quickly.

By October 1, the total number of transactions had been reduced to 1,022 and the total number of reserves had been reduced to 305, less than half the number in September. The shorter reserve list meant less staff time spent in searching. By November 1, the reserve list reached a low of 286 requests. However, by February 1, this number had increased to 356.

RESULTS

After six months of working with the microcomputer, the staff feel that it is worth continuing as a permanent part of the procedure. The reserve lists have proved to work much more efficiently, and the time saved by not having to keep a daily log for items sent and returned is significant. The

AUTHOR'S LAST NAME:
AUTHOR'S FIRST NAME:
TITLE:
CALL NO.: CUTTER NO.:
IMPRINT:
PATRON: PHONE:
BORROWING LIBRARY: PLACE:
LENDING LIBRARY:
DATE REQ.: DATE SENT: DATE DUE: DATE RET.:
STATUS:
DATE ENTERED:
NOTES:

Fig. 1. Interlibrary Loan Request Form.

use of the PFS: Report program for keeping statistics requires a lot of paper, in that a ten-page report is printed out each month. But, the process, which took four hours a month previously, now takes one hour.

Once all of the records were input, the inputting time could be handled by the same interlibrary loan staff used prior to computerization. Fortunately, there have been no mechanical failures thus far. Backup copies are made daily as a precaution. Paper copies of requests are still kept and filed. Ultimately, this too can be eliminated.

An added aspect of database searching is the ability to search for items by any field. The PFS: File does a random access search for the first field, which in this case is the author's last name. The other files are searched linearly, which takes longer—anywhere from two to five minutes.

Most items can be found by the author's last name, and the item is called up on the screen immediately. However, if the author's last name is misspelled, then it will not be found, and the item will have to be searched by title, which will take longer. In this regard, PFS programs were easy to learn and were able to handle the size of the files involved. However, much larger files would slow down response time and make copying difficult.

An additional benefit of the PFS: File has been the printing of bills of lading for shipping materials by Purolator, which is the contractor for the Interlibrary Delivery Service of Pennsylvania. The bills of lading had previously been printed by hand or typed and took as much as half an hour to fill out on delivery days. This is done much more quickly by computer. One difficulty that arose was that the forms changed shortly after the library started printing them by computer. This took several months to resolve.

CONCLUSION

The introduction of the microcomputer to the interlibrary loan department has been judged a success both by the staff and by the libraries being served. The additional benefit of being able to have a more accurate picture of interlibrary loan activity has helped in managing for increased efficiency. ■■

The DAPPOR Answer Evaluation Program

Bert R. Boyce, David Martin, Barbara Francis, and Mary Ellen Sievert

INTRODUCTION

DAPPOR (Drill and Practice Program for Online Retrieval) is a series of computer programs and a database whose purpose is to provide a student with drill at entering command protocols in the proper syntax for any of the four major vendors of online databases. In principle, the programs could be used for drill in any vendor's command language, but the DAPPOR database currently under development is only for the National Library of Medicine's (NLM) Elhill system, Lockheed's Dialog System (LRS), System Development Corporation's (SDC) Orbit System, and the Bibliographic Retrieval System (BRS).

The project grew from a perceived need to use computer-assisted instruction to supplement classroom teaching on the use of online retrieval systems. The available packages did not seem to meet adequately the needs of library school students. The vendor's classroom-instruction accounts certainly make the use of their systems economically feasible for practicing interactive searching. However, the use of these systems for learning command syntax seems expensive, and a local instructional package seems more suited for this area. Such a package as TRAINER,[1] the University of Pittsburgh's emulator, would provide low-cost interaction. Unfortunately, the cost and effort of converting TRAINER to IBM equipment, which is common in many library schools, outweighs the savings generated by its use. It is, of course, also the case that emulators are quickly outdated in this rapidly changing environment.

The Individualized Instruction for Data Access System developed at Drexel,[2] while providing interactive instruction, is Dialog specific, is designed for the end user rather than library science students, and is database dependent.

It would seem that when in-depth classroom instruction is provided, and when educational accounts are available to make realistic practice searching a possibility, the most advantageous use of CAI will be to reinforce basic command syntax for the student.

DAPPOR is a tool for use in conjunction with regular instruction in the command protocols of literature searching systems and is not designed as an independent learning package. It makes no attempt to teach search strategy, question analysis, or iterative techniques. Its sole purpose is to reinforce the learning of the proper syntax in any vendor's system.

In this paper we wish to present the general structure of DAPPOR and its database and, particularly, to describe the coding process for answer definitions and the verification program that will reduce correct answers to null (empty) strings.

THE DAPPOR DATABASE

The DAPPOR database consists of fourteen classes of questions that represent common functions of the systems which specifically deal with the search process. The selected classes are shown in table 1. Informational operations not specifically associated with the search process (e.g., time) are not covered. The assumption is that the searcher will not have the same immediate need for these functions as she or he will for those associated with actual interactive search. The syntax for searching, however, should be impressed firmly in the searcher's memory.

Each of these classes contains a series of at least ten questions that require the user to enter a statement in the proper syntax. DAPPOR will choose a class and a question from that class at random and present it to the user. The user's response will then be evaluated, and, if it is correct, the proba-

---

Bert R. Boyce is a faculty member of the School of Library and Information Science at Louisiana State University, Baton Rouge. David Martin, Barbara Francis, and Mary Ellen Sievert are associated with the Department of Information Science of the University of Missouri at Columbia. This publication was supported in part by NIH Grant LM03980 from the National Library of Medicine.

Table 1. DAPPOR Question Classes

Class	Name	Function
1	Local printing	
2	Remote printing	
3	Temporary and permanent saves	
4	Off searches and SDIs	
5	Working storage adjustments	
6	Numerical field limitation	
7	File change and sign off	
8	Search I	Simple and nested Boolean function entry with index terms and working storage sets.
9	Search II	Adjacency.
10	Search III	Truncation.
11	Search IV	Field label qualification.
12	Search V	The creation of lists of system-generated index entries.
13	Search VI	The use of system-generated lists.
14	Search VII	String searching.

bility of repeating that class of question will be reduced as well. If the response is not correct, only the probability of the question will be reduced, and not the probability of the class. Thus, classes where the user answers incorrectly will be presented more often to the user. After each question DAPPOR will return an evaluation of correctness and a single preferred form of the correct answer.

ANSWER EVALUATION

Each user response is in the form of a character string that will be compared to an answer definition. A key problem in DAPPOR is the evaluation of responses to the questions DAPPOR poses to its users. Because of the flexibility of the vendors' programs, the number of possible correct answers for a specific drill question, while finite, is unfortunately large. This precludes character-string matching as a viable evaluation strategy. Therefore the answer-evaluation program described below was constructed.

Each question has associated with it an answer definition. This definition contains at least one character and no more than 4,096 characters. Concatenation of characters is implied by juxtaposition. The characters "]", "[", "}", "{", "~", "&", "|", and "%" have special meanings and cannot be used in a response. They are integral parts of an answer definition.

The answer-definition program operates by recursively reducing a response by the application of an answer definition and reduction rules. In other words, the program compares the response with the answer definition, eliminating from both those elements that match and eliminating from the answer definition any other elements required by the rules. The response is correct if the program can reduce the answer to a null (empty) string and the answer definition to optional expressions only. The following six sections provide a formal description of construction and reduction of answer definitions.

The evaluation process involves a comparison of the response character string with a character string called a DEFINITION. The syntax rules for forming a DEFINITION and the evaluation rules for comparing the strings are specified below.

Preliminary Editing of a DEFINITION

Each DEFINITION is stored in a variable-length record together with its corresponding prompt and preferred response. The maximum length of a record is 4,096 characters. This fact cannot be used to determine in advance whether or not information will fit in a record, however, since a DEFINITION is actually stored in a list format in an array. When the DEFINITION character string is converted to list form,

1. a lowercase letter is replaced by the corresponding uppercase letter;
2. a substring of two or more consecutive blanks is replaced by a single blank;
3. a substring of two or more consecutive tildes (˜) is replaced by a single tilde;
4. a substring of two or more percent signs (%) is replaced by a single percent sign;
5. any blanks immediately following a logical OR symbol (|) are not stored;
6. any blanks immediately following a logical AND symbol (&) are not stored; and
7. trailing blanks are not stored.

Special Characters, SSETs, and PSETs

Within a DEFINITION any of the characters "}", "{", "]", "[", "˜", "&", or "|" is called a *special character*. A nonempty substring of a DEFINITION is called an SSET (string set) if it does not contain any special characters and if the substring cannot be made any longer without including a special character. Thus the SSETs in the DEFINITION "{PRINT | PRT}" SKIP 10 ˜OFFLINE" are "PRINT," "PRT," "SKIP 10," and "OFFLINE." Note that blanks are treated just like any other nonspecial character.

"[,AU,TI]" is an example of a PSET (permutation set). A permutation set is a substring of a DEFINITION that consists of a set of square brackets enclosing a nonempty string where (1) none of the characters in the string is a special character and where (2) the first character of the string occurs at least twice within the string but is not the same as the last character of the string and does not occur in two consecutive positions of the string. The first character of the string in a PSET is called the *delimiter* of the PSET. From the definition of a PSET, it follows that the string in a PSET is partitioned into two or more nonempty substrings by the PSET delimiter. Each of these substrings is called a PSET element.

DEFINITION Syntax

The rules given below specify how SSETs and PSETs are combined to form DEFINITIONS. The compound symbol "=:" means "equals by definition." The compound symbol "| |" is used to indicate concatentation. However, when a DEFINITION is actually written, concatenation is indicated by juxtaposition; the "| |" symbol never appears in a DEFINITION. Note that the definition of an OPERAND is circular since OPERAND is defined in terms of a GROUP, which is defined in terms of an EXPRESSION, which is defined in terms of an OPERAND. In constructing a DEFINITION, this means that a GROUP can contain nested GROUPs. There is no limit on the level of nesting of GROUPs other than that imposed by the maximum length of a DEFINITION that can be stored.

Significance of DEFINITION Operators

Square brackets are used in definitions to delimit PSETs. Braces are used to delimit GROUPs. The remaining three special characters ("˜", "&", and "|") are used as operators. Even though there are only three operation symbols, there are in fact five operations that can be used in a DEFINITION to combine UNITS (i.e., SSETs and PSETs).

One of the *extra* operations is the concatenation operation, which has no explicit operator. The other extra operation results from the fact that the logical OR symbol denotes two different operations, depending upon context. When used outside a GROUP, the logical OR symbol represents a logical OR operation. However, when used inside a GROUP, the logical OR symbol represents the logical EXCLUSIVE OR operation.

While the logical AND symbol does represent the logical AND operation, the tilde ("˜") does not represent the logical NOT operation. It is used in a DEFINITION to denote the optional UNITEXPRESSIONs and optional GROUPs.

With the five operations defined, a general description of a DEFINITION can be given. A DEFINITION is the logical OR of one or more ANSWERs. An ANSWER is the concatenation of one or more TERMs. A TERM is either a UNIT, an optional UNIT, or a GROUP. And finally, a GROUP is one or more GROUPs, UNIT-

EXPRESSIONs or optional UNITEXPRESSIONs combined with the logical AND and/or EXCLUSIVE OR operations. (See figure 1.)

When a GROUP contains three or more OPERANDs combined by the "&" and "|" operators, the priority of operations is left to right. Thus each operator combines the operand following with an implicit GROUP containing all of the operands that precede the operator. Thus, for example, {A | B&C | D} is equivalent to {{{A | B}&C} | D} and {A | { B&C | D} | E} is equivalent to {{A | {{B&C} | D}} | E}.

Finally, it is important to note that the concatenation of UNITs occurs in two contexts. If UNITs are concatenated outside a GROUP, each UNIT is a TERM. If UNITs are concatenated inside a GROUP, the concatenation forms a UNITEXPRESSION. This distinction means that "A¯B" would be a valid DEFINITION while "{A¯B}" would not be valid. (A TERM can be an optional UNIT while a UNITEXPRESSION cannot contain an optional UNIT.) This distinction also means that in "¯A[,B,C]" only "A" is optional while in "¯{A[,B,C]}", "A[,B,C]" is optional.

Preliminary Editing of a Response

Before a response is compared to a DEFINITION, the following editing will be performed on the response:

1. a response containing only blanks is replaced with the null string;
2. any lowercase letter is replaced by the corresponding uppercase letter;
3. trailing blanks are deleted; and
4. any substring of two or more blanks is replaced with a single blank.

Division of UNITEXPRESSIONs

Evaluating a response requires a *comparison* of the edited response to a DEFINITION. This comparison involves a process called *division*. Let C be any character string and let S be any set of character strings. Then C is said to be *divisible* by S if there exist strings X and Y such that $C = X \mid\mid Y$ and X is an element of S. If X is the longest element of S such that $C = X \mid\mid Y$, then X is called the *quotient* and Y is called the *remainder* in the division of C by S.

If SSETs and PSETs are associated with sets of character strings, then division of character strings by UNITs will follow

```
1. UNIT =: SSET
   Example: A,B
        =:PSET
   Example: [,A,B]
2. UNITEXPRESSION =: UNIT
                  =: UNITEXPRESSION | | UNIT
   Example: [,A,B]C
3. OPERAND =: UNITEXPRESSION
           =: ¯UNITEXPRESSION
           =: GROUP
   Example:¯[,A,B]C
4. EXPRESSION =: OPERAND
              =: EXPRESSION&OPERAND
              =: EXPRESSION | OPERAND
   Example: [,A,B]C&D | E
5. GROUP =: {EXPRESSION}
         =: ¯{EXPRESSION}
   Example: {[,A,B]C&D}
6. TERM =: UNIT
        =: ¯UNIT
        =: GROUP
7. ANSWER =: TERM
          =: ANSWER | | TERM
8. DEFINITION =: ANSWER
              =: DEFINITION | ANSWER
```

Fig. 1. Answer-Definition Elements.

from the preceding definition. The set associated with an SSET is the set of all character strings that can be obtained by considering the percent sign as an optional blank. Thus, for example, the set associated with the SSET "PRINT" contains the single string "PRINT" while the set associated with the SSET "A%B%" contains the strings "AB", "A B", "AB ", and "A B ". The SSET "A%B%" divides the character string "AB DC" but does not divide the character string "CAB DC". Note that when the "AB DC" is divided by "A%B%", the quotient is "AB ", and the remainder is "DC". The quotient is not "AB" since "AB " is longer than "AB".

The set associated with a PSET is the set of all strings that consist of permutations of the PSET elements where consecutive elements are separated by the PSET delimiter. As an example, the set of strings associated with the PSET "[,A,B,C]" consists of "A,B,C", "A,C,B", "B,A,C", "B,C,A", "C,A,B", and "C,B,A". Thus when the "A,C,B,X,Y,Z" is divided by "[,A,B,C]", the quotient is "A,C,B" and the remainder is ",X,Y,Z".

Division of a character string by a UNITEXPRESSION requires a definition of division by a concatenation of UNITS. When $U(1)$, $U(2)$, $U(3)$, . . . represent any sequence of UNITs, then for any positive integer n, a character string C is divisible by $U(1) \mid\mid U(2) \mid\mid \ldots \mid\mid U(n) \mid\mid U(n+1)$ if C is divisible by $U(1) \mid\mid U(2) \mid\mid \ldots \mid\mid U(n)$ and the remainder of that division is divisible by $U(n+1)$. The quotient in this case is the concatenation of the two quotients, and the remainder is the remainder of the last division.

Response Evaluation

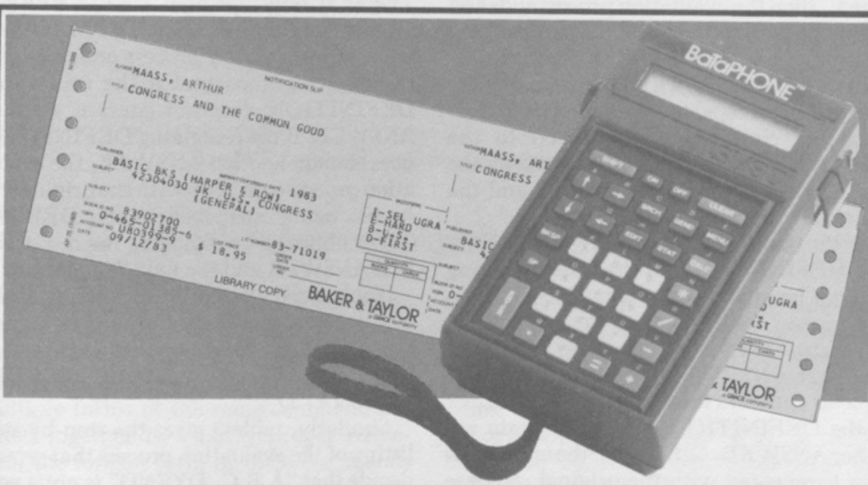
As can be seen from the section "Significance of DEFINITION Operators," any DEFINITION can be expressed in such a manner that each GROUP contains no more than two operands. Thus in the description of response evaluation that follows, it will be assumed that each GROUP has no more than two operands.

Note that a DEFINITION contains a sequence of UNITEXPRESSIONs. The evaluation of a response begins by locating the first UNITEXPRESSION within the first TERM of the first ANSWER, such that the response is divisible by the UNITEXPRESSION. If such a UNITEXPRESSION is found, the process continues with the remainder of the division and a *reduced* DEFINITION. If the UNITEXPRESSION found was not within a GROUP, then the first TERM must be just the UNITEXPRESSION found or the UNITEXPRESSION found preceded by a "|". In this case the reduced DEFINITION is obtained by removing the first TERM.

If the UNITEXPRESSION found is within a GROUP, obtaining the reduced DEFINITION is more complicated. In the procedure that is described below, it is assumed that whenever all of the operands of a GROUP are removed from the DEFINITION, the enclosing braces are also removed. It is also assumed that a "|" operator is removed whenever the UNITEXPRESSION or GROUP that it precedes is removed. Remembering that no GROUP has more than two operands, the reduced DEFINITION is obtained as follows:

1. Remove the UNITEXPRESSION found.
2. If the UNITEXPRESSION found was an operand in a GROUP with two operands, remove the "&" or "|" operator and, if the operator removed was the "|", remove the other operand also.
3. Locate every operand that is combined by an "|" operator with a GROUP that contained the UNITEXPRESSION found. Remove all of these operands and "|" operators.
4. If any GROUP has been left with a single operand and an "&" operator, remove the "&".

The procedure described above is repeated until (1) the remainder in the division of the response string is null, (2) all of the first TERM has been removed from the DEFINITION, or (3) no UNITEXPRESSION in the first TERM will divide the response string. If the remainder of the division of the response string is null, then the evaluation process ends and the response was correct if the remaining portion of the first ANSWER contains only optional UNITEXPRESSIONs (i.e., UNITEXPRESSIONs preceded by "~" or contained in GROUPs preceded by "~"). If the re-


BaTaPHONE
LIBRARY COPY
BAKER & TAYLOR

maining portion of the first ANSWER contains a UNITEXPRESSION that is not optional, then the evaluation process ends and the response was incorrect if the remaining portion of the DEFINITION does not contain another ANSWER. Otherwise, the entire process is repeated with the original response string being compared to the DEFINITION obtained by discarding the remainder of the first ANSWER and the following "|".

If the first TERM has been removed from the DEFINITION and the response string has not been reduced to a null string, then the process continues with the next TERM if it exists. If there are no more TERMs in the first ANSWER, then the evaluation process ends and the response was incorrect if the DEFINITION does not contain another ANSWER. Otherwise, the entire process is repeated with the original response string being compared to the DEFINITION obtained by discarding the leading "|".

Finally, if the response string has not been reduced to a null string and no UNITEXPRESSION of the first TERM will divide the response string, then the evaluation process continues with the DEFINITION obtained by removing the first TERM if only optional UNITEXPRESSIONs remain in the first TERM. Otherwise, the evaluation process ends and the response was incorrect if the remaining DEFINITION does not contain another ANSWER. If the remaining DEFINITION does contain another ANSWER, the evaluation process continues with the original response being compared to the DEFINITION obtained by removing the rest of the first ANSWER and the next "|" operator.

Table 2 gives a step-by-step listing of the evaluation process that would conclude that "ACE" is a valid response according to the DEFINITION "A{B | C&D | E} | {E&C&A}¯D".

Similarly, table 3 gives the step-by-step listing of the evaluation process that would decide that "A,B,C, DYEND" is not a correct response when the DEFINITION is "A,[,B,C,]{X&Y}END".

THE CODING PROCESS

To exemplify the coding process, three samples will be given. Each will start with the question as it would be viewed on the

Table 2.

<table>
<tr><th>Response</th><th>Definition</th></tr>
<tr><td>ACE</td><td>A⃰{{{B | C}&D} | E} | {{E&C | &A}¯D
The initial "A" in the response is found immediately in the definition and removed from both.</td></tr>
<tr><td>CE</td><td>{{{B | C⃰}&D} | E} |{{E&C}&A }¯D
(1) The "C" in the response is found and removed from both. (2) The operand "B" is removed from the answer definition since it is to the left of the " | " preceding the "C" as is the " | ", which itself is also removed. (3) The "D" is also removed since "D" is now a single operand. (4) The " | " and the "E" are removed as operator and operand since they are in a group containing the expression found.</td></tr>
<tr><td>E</td><td>{{D⃰}} | {{E&C}&A}¯D
Since "D" remains to the left of the " | " there is no reduction. The program now begins with an attempt at reduction to the right of the " | ".</td></tr>
<tr><td>ACE</td><td>{{E&C}&A⃰}¯D
The "A" is removed, and, since a single operand remains, the "&" is removed as well.</td></tr>
<tr><td>CE</td><td>{{E&C⃰}}¯D
The "C" is removed, and, since a single operand remains, the "&" is removed as well.</td></tr>
<tr><td>E</td><td>{{E⃰}}¯D
The "E" is removed from both.</td></tr>
<tr><td>Null</td><td>¯D⃰
Since only an optional unit remains in the answer definition, the response is correct.</td></tr>
</table>

Table 3.

Response	Definition
A,B,C,DYEND	A,[,B,C,D]{X&Y}END The initial "A," is found and removed from both.
B,C,DYEND	[,B,C,D]{X&Y}END The "B,C,D" meet the requirements of the PSET [,B,C,D] and both are removed.
YEND	{X&Y}END The "Y" is removed, and, since a single operand remains, the "&" is removed as well.
END	{X}END Since the response string does not begin with "X", the response is incorrect.

terminal by the student. This is followed by the allowable responses and the answer definition, the character string that represents all of the possible correct answers. When multiple forms of the same command are ORed together to avoid improper reduction, the longest form of those commands with the same initial characters must be entered first.

Figure 2, a question from class 7 on changing from one database to another, is an example of an easy coding process.

On LRS, the command is either the full word "begin" or the abbreviation "b" followed by the file number "30", either with or without a space between. The special character "%" represents this optional blank.

SDC and NLM are straightforward with the "file" command, a space shown with a b̸ and the file name.

BRS requires the " . . " followed by the command "change" or its abbreviation "c", a slash mark "/", and the four-letter file name. The "/" can be separated from the command and the file name by a blank or no blank, thus the "%" for this optional blank.

A more difficult problem was encountered in coding the adjacency search questions, in class 9, shown in figure 3. In both figure 3 and figure 4, we have not shown all the possible combinations of correct responses. For example, any of the {sel} commands can be used with any of the (w) combinations. To represent the many search commands for LRS, the "sel" is used as a shortcut in coding. Such macrocodes are translated to full form by the input editor program and are often used in the coding process. Following the "{sel}" command is an optional blank "%" and the three search terms separated by the proximity operator (w). Spaces may appear on the outside and on the inside of the parenthesis; therefore %(%w%)% goes between the search terms.

For SDC, the command "find" or its abbreviation "fd" is optional, indicated by the "~" preceding each. There are several adjacency operators that could be used between the search terms: "adj"; "(w)"; "(0)"; "w/0"; and "()". All but the first have op-

Question:
You are in an unknown database and desire to enter a chemical database to conduct a new search. Enter the appropriate command.

LRS: The file number is 30
Responses:
? b 30
? b30
? begin 30
? begin30
Answer definition
{begin | b} % 30

SDC: The file name is CAS77
Responses:
ss8/c
USER: file cas77
Answer definition:
fileb̸CAS77

BRS: The file name is CHEM
Responses:
_: . . change/chem
_: . . c/chem
_: . . . change / chem
_: . . c / chem
Answer definition:
. . {change | c} % / % chem

NLM: The file name is chemline
Responses:
ss8/c
USER: file chemline
Answer definition:
fileb̸chem¯line

Fig. 2. Sample of Class 7.

Question:
You wish to find material on the specific phrase "computerized axial tomography." Use adjacency operations.

```
LRS:
    Responses:
      ? selectsteps computerized (w) axial (w) tomography
      ? select steps computerized (w) axial (w) tomography
      ? select computerized ( w ) axial ( w ) tomography
      ? sstepscomputerized(w)axial( w )tomography
      ? s steps computerized (w) axial (w) tomography
      ?scomputerized( w ) axial(w)tomography
    Answer definition:
      {sel}* % computerized % ( % w % ) % axial % ( % w % ) % tomography
SDC:
    Responses:
      SS1/C
      USER:   find computerized adj axial adj tomography
              fd computerized w/0 axial w/0 tomography
              computerized (w) axial (w) tomography
              computerized(0)axial(0)tomography
              computerized () axial () tomography
    Answer definition:
      ~{findb̸ | fdb̸}computerized{sdcadj}**axial{sdcadj}**tomography
BRS:
    Responses:
      1__: computerized adj axial adj tomography
      1__: computerized adj axial tomography
    Answer definition:
      computerizedb̸adjb̸axialb̸~{adjb̸}tomography

 *Macrocode for: {selectb̸steps | select | ssteps | ss | sb̸steps | s}
**Macrocode for: {{b̸ADJb̸ | % ( % w % ) % | % ( % 0 % ) % } | {b̸w % / % 0b̸ | % ( % ) % }}
```

Fig. 3. Sample of Class 9.

tional spaces involved. These operators were assigned to the macro "SDCADJ" instead of writing out this string each time. A blank (b̸) cannot be counted as significant if it follows an "|" in the coding, so it was necessary to divide the five choices into two groups to get the b̸ as the first character in the string.

For BRS, the only operator that applies is the "adj". In this case it is necessary between the first two terms but is optional between the second and third terms as the BRS system defaults to the same operator if there is a space between any terms after the first two.

On NLM no proximity searching is possible. Similar questions on NLM and SDC using string searching are also included in class 14, Search VII.

The two most difficult questions to code were those on print commands, both local and remote. It was necessary for coding purposes to narrow the options for some systems. Thus, for SDC and NLM it was necessary to restrict the use of field labels. For example, in SDC the field labels "AU" "TI" "SO" can be arranged in any sequence and can be separated by blanks, commas, or both commas and blanks, or by the number of items to be printed, or by other items in the command.

Figure 4 is an example of one of the easier offline print questions, from class 2. LRS is fairly simple, with optional blanks between almost every item.

On SDC the three required elements (print or prt, offline, and storad) are easy to handle, but allowing for the three options for numbers of documents and the optional search set number and the optional indented format complicates the coding. For NLM the macrocode "NLM" was set up to indicate the possible optional formats that would otherwise need to be written out each time.

On BRS the four options for bibliographic format are not such a problem because of the fact that they must appear after the search set number and before the first "/". The [,au,ti,so] is a PSET meaning that the three terms can come in any order but must be separated by commas, indicated by the comma's coming first in the brackets. Because there are two "doc =" options, they are grouped with an "|" to show either can be used. The "doc =" string and the "id =" string can be present in either order, this being indicated by the "&" between

Question:
Your search at Statement 5 has resulted in 300 postings and your patron, Brown, says to print them all offline in bibliographic citation format. Enter the proper command after the prompt.

```
LRS: Responses:
        ? print 5/3/1-300
        ? pr5 / 2 / 1 - 300
        ? print5/ 2/ 1-300
        ? pr 5/3/1-300
     Answer definition
        {print | pr}%5%/%{3 | 2}%/%1-%-%-300
SDC:
     Do not use field labels in your answer. Ensure that you will not be prompted for a mailing address.
     Responses:
        SS6/C
        USER: print offline storad 300 ss 5 indented
              print storad offline 1-300
              prt-1-300 storad offline
              prt offline storad.
     Answer definition:
        {print | prt}ƀ{offlineƀ&storadƀ&¯{300ƀ | 1%-%300b| -%1%-%300ƀ}
        &s̄sb/5ƀ&îndentedb/} | {print | prt}-%1%1%-300ƀ{offlineƀ&
        storadƀ&s̄sb/5ƀ&îndentedb/}
BRS:
     Responses:
        6: . . printoff 5 bibl/doc = 1-300/id = brown
           . . printoff 5 f3/id = brown/doc = all
           . . po 5 f2 / doc = 1-300 / id = brown
           . . po 5 au,ti,so / id = brown / doc = all
     Answer definition
        . . {printoff | po}ƀ5ƀ{bibl | f3 | f2 | [,au,ti,so]){{%/%doc% = %all |
        %/%doc% = %1-300}&%/%id% = %brown}
NLM:
     Do not use field labels in your answer.
     Responses:
        SS6/C
        USER: Print offline
              prt offline ss 5 indented
              prt 300 offline compressed
              prt offline ss 5 standard
              print 300 compr offline
              print offline compact
     Answer definition:
        {print | prt}ƀ{offlineƀ&¯ssƀ5ƀ&¯300ƀ&¯{nlm*ƀ}}
*Macrocode for: {indentedƀ | compressedƀ | compressƀ| comprƀ | compactƀ | standardƀ | SDƀ}
```

Fig. 4. Sample of Class 2.

them. This is all enclosed in {}s because, in the coding, an "&" must appear within {}s.

CONCLUSIONS

Our review of the command syntax of the four systems shows that the degree of flexibility available to the searcher in entering commands is, in fact, quite impressive. While this flexibility is of value to the online searcher, it makes the evaluation of student input quite complex.

Simple string matching was rejected as a viable solution early in the effort. The answer-evaluation program described above meets the needs of a drill-and-practice program on command syntax quite well. The necessary coding of answer definitions, while time-consuming, is no more so than would be the enumeration of all the possible correct forms.

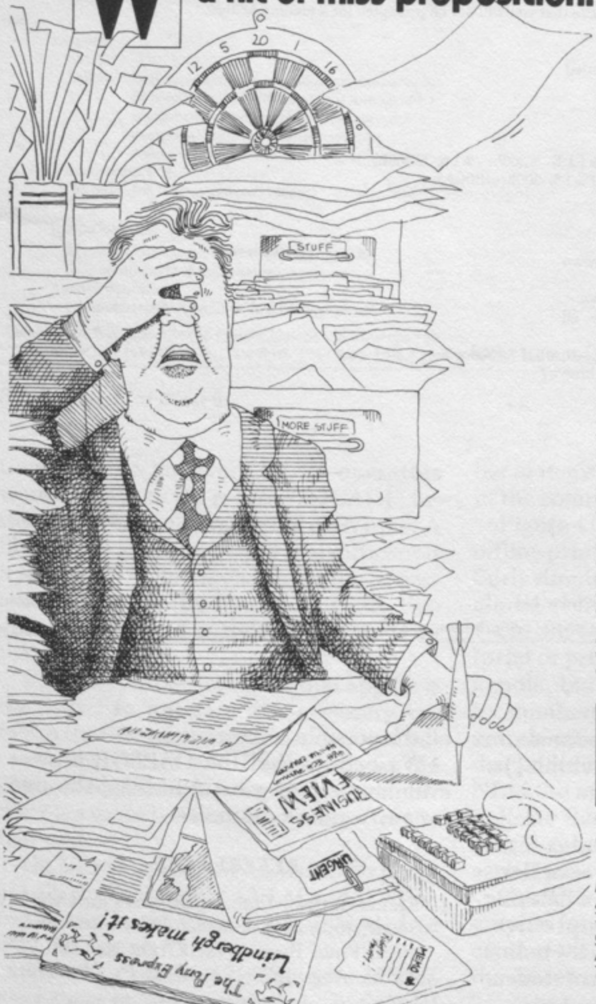
We believe that the DAPPOR answer evaluator is quite general and could be used for any vendor's protocols.

REFERENCES

1. Elaine Caruso, *Training Modules for Users of Scientific and Technical Information Services,* Final Report NSF Grant No. DSI 77-26525, Aug. 1979 (Pittsburg, Pa.: Pittsburg Univ.).
2. Charles T. Meadow, Thomas T. Hewett, and Elizabeth S. Adversa, "A Computer Intermediary for Interactive Database Searching, I. Design," *JASIS* 33, no.5:325–32 (Sept. 1982); Charles T. Meadow, Thomas T. Hewett, and Elizabeth S. Adversa, "A Computer Intermediary for Interactive Database Searching, II. Evaluation," *JASIS* 33, no.6:357–64 (Nov. 1982).

■■

With MINISIS, Database Management isn't a hit or miss proposition.



Database management (the exacting art of finding things in the Information Age) is becoming increasingly important to all manner of organizations that have to maintain, manipulate and retrieve large volumes of information.

The famous Cranshaw Method (pictured on the left) is fast becoming a thing of the past. But, there are a lot of database systems out there that can make you feel like Cranshaw - with a keyboard instead of a dartboard.

MINISIS, from Systemhouse, is an extremely powerful, extremely easy way of finding things. It doesn't matter what things you want on your database - journals, books, real estate listings, videotapes, people, case histories, reports, machine parts, companies, legal cases, racing tips - anything. Because MINISIS is a "relational" database system, it will let you use data in ways you may not even have had in mind when you started. Without any computer background, MINISIS users can create new applications, often in a single day. And if what you get isn't exactly what you wanted or if you've changed your mind, you simply change the database.

Moreover, MINISIS has everything you need to develop your own records management system or information retrieval application, including a very powerful query capability with keyword and text searching and a super report generator that can produce anything from a catalog card, to a form letter, to a telephone directory. It even has processors to convert your existing databases into MINISIS format.

Running on the HP3000 family of minicomputers, MINISIS is being used by 80 organizations in 24 countries to manage their information. Universities, special libraries, major corporations, museums, documentation centres and more.

So if you're looking for a comprehensive, cost-effective, proven method of finding things, we think that your search, unlike Cranshaw's, is over.

Call us.

SYSTEMHOUSE

We don't make computers...
We make them work.

MINISIS Product Managers

Systemhouse Inc.*	Systemhouse Ltd.
1655 North Fort Myer Drive	2827 Riverside Drive
5th Floor,	Ottawa, Ontario
Arlington, Virginia 22209	K1V 0C4
703-276-0500	613-526-0670

Los Angeles • San Francisco • Chicago • Washington, D.C. • New York • Boston
Vancouver • Regina • Winnipeg • Toronto • London • Ottawa • Montreal • Halifax

*d/b/a SHL Inc. in Connecticut, Illinois, New Jersey, Texas and Massachusetts, and Systemhouse Computer Services Inc. in New York.

Currie-Irving Advertising, Ottawa-Toronto

News and Announcements

Microcomputer Software for Libraries

Capital Systems Group (CSG) has announced two new software packages for the library and publishing market. NEWSDEX and BOOKDEX are now available for the IBM PC and compatible microcomputers. NEWSDEX generates indexes for newspapers and periodical literature; BOOKDEX prepares back-of-the-book indexes. Both packages were developed by CSG and are based on indexing programs originally designed by the late Theodore H. Hines. They enable the indexer to post, sort, and subordinate entries under chosen subject headings; they minimize the chores of keying and editing while maintaining the quality of indexing achieved only through intellectual effort.

Other library applications software marketed and distributed by CSG include CHECKMATE and GOLDEN RETRIEVER, the serials and bibliographic database managing systems developed by CLASS, and SCIMATE, ISI's Universal Online Searcher and Personal Text Manager.

For further information contact Renee Schick or Carol Cochran at Capital Systems Group, 11301 Rockville Pike, Kensington, MD 20895 or phone (301) 881-9400. ■■

A Microcomputer Acquisitions System

Bib-Base/Acq features a full-screen editor for record input and uses variable-length records containing MARC-tagged variable-length fields. All author and title fields are indexed, as well as the call number and order number. The database may be searched using any of these elements. As records are added, deleted, or modified, the indexes are updated.

Based on this database, Bib-Base/Acq can be used (1) to determine if a particular title is in the database and what its current status is; (2) to create lists of records selected from the database (e.g., orders outstanding after six months); (3) to produce orders to vendors; (4) to check-in items as they are received; and (5) to generate encumbered and expended fund reports.

Bib-Base/Acq runs on the IBM-PC, Zenith Z100, and other microcomputers using the MS-DOS or CP/M-86 operating system. It requires two disk drives,-192K of memory, and a printer. Bib-Base/Acq costs $895. The detailed 120-page operating manual ($30) and a demonstration version of the program ($45 with manual) are available separately. For more information, contact Small Library Computing, 837 Twining Rd., Dresher, PA 19025. ■■

ILL Software Package

MacNeal Hospital's Health Sciences Resource Center has released a software package, F.I.L.L.S. (*F*ast *I*nter *L*ibrary *L*oans and Statistics), that saves time in the interlibrary loan function. It also produces a variety of relevant reports. It is geared to small libraries that utilize the IBM PC.

F.I.L.L.S. does not require a database management software package and costs $295.

In addition to the IBM-PC, *F.I.L.L.S.* also can be used on the IBM-XT and the IBM-M300. It has been designed to print on A.L.A.-approved, four-part, continuous pin-feed library loan forms and will support most printers which are capable of printing 10 characters per inch.

Further information on *F.I.L.L.S.* is available by contacting Rya Ben-Shir, Health Sciences Resource Center, MacNeal Hospital, 3249 S. Oak Park Ave., Berwyn, IL 60402; 312/795-3089. ■■

Two Midwestern Libraries Become Associate Members of RLG

The University of Illinois at Urbana-

Champaign and the University of Wisconsin–Madison have become associate members of the Research Libraries Group in order to participate in RLG's East Asian Studies Program. They join twelve other libraries already taking advantage of RLG's online database of East Asian vernacular catalog records. A grant to RLG from the Lilly Endowment in Indianapolis supports the start-up and initial operations costs for both libraries.

The decision of the two libraries to join further strengthens a program initiated by RLG in 1979 to help devise strategies for national planning in the East Asian library field. The program aids such planning with computer-supported bibliographic processing, cooperative collection development, and the development of special databases related to East Asian studies. Illinois and Wisconsin both have East Asian collections in excess of one hundred thousand volumes and support graduate and undergraduate programs in East Asian studies.

To achieve national bibliographic control of East Asian vernacular material, RLG has expanded its automated library information system—Research Libraries Information Network (RLIN). Enhancements to RLIN allow materials in Chinese, Japanese, and/or Korean to be stored and retrieved at library computer terminals as readily as Western-language publications. The twelve libraries now using the RLIN CJK enhancements are Brigham Young University, Brown, Columbia, Cornell, the Hoover Institution at Stanford, the Library of Congress, the Los Angeles County Public Library, Princeton, the Universities of Michigan, Minnesota, and Pennsylvania, and Yale.

In late February, just six months after the East Asian enhancements to RLIN were implemented, the ten-thousandth vernacular record was entered into the database. Program members are now entering CJK records at the rate of more than twenty-seven hundred per month.

Illinois and Wisconsin will be the first institutions to receive the new smaller, tabletop versions of the RLG CJK terminal cluster controller, released last month by RLG and Transtech International Corporation. This recent change in hardware has reduced the cost of an RLG CJK terminal by 28 percent. ■■

OCLC's LS/2000

OCLC has announced that East Carolina University has acquired software to mount an OCLC LS/2000 local library system in its J. Y. Joyner Library, Branch Music Library, and Health Sciences Library on the university's campus at Greenville, North Carolina.

OCLC will provide LS/2000 software and support services—training, profiling, installation, and software maintenance—for East Carolina University's local library system.

The database load is scheduled to begin July 15, 1984. SOLINET will create from East Carolina University's subscription tapes a local database of approximately 250,000 records. SOLINET will insert unique identifiers in the records using the 949 field and will generate a print tape of these identifiers for a third-party vendor to use in printing bar code labels.

East Carolina will initially use the LS/2000 Circulation and Bibliographic Control (cataloging, editing of cataloging records, and authority control) Subsystems, with activation of the Catalog Access Subsystem (online catalog) to occur later. The system will start up with twenty-one terminals and will operate on a minicomputer used by the three libraries. The LS/2000 system will have a printer port link to the OCLC Online Union Catalog.

East Carolina University's Joyner Library, Branch Music Library, and Health Sciences Library serve a total enrollment of 13,357 students and 864 faculty members. The libraries have combined holdings of 525,000 titles and 745,000 bound volumes. Last year they had more than 297,000 circulation transactions. East Carolina University has been an OCLC member since 1975.

The University of Kentucky has also acquired an OCLC LS/2000 local library system. The system is intended for the Margaret I. King Library and Medical Center Library on the university's campus at Lexington and, ultimately, the twelve community colleges that comprise the University of Kentucky Community College System.

The University of Kentucky is the first Association of Research Libraries institution to acquire an LS/2000 system. The system being installed initially includes a Data General MV 10,000 minicomputer and twenty-seven terminals for circulation, bibliographic and authority control, and limited public access. Later phases will include terminals for increased public access.

In preparation for the database load, which is scheduled to begin in August 1984, OCLC will extract the university's bibliographic records from OCLC archive tapes, consolidate them, and convert headings to the proper AACR2 form using the current library of Congress Name-Authority File.

The University of Kentucky Libraries have 1.8 million volumes and approximately thirty-five thousand current serials.

Under terms of the agreement, OCLC will provide the University of Kentucky with LS/2000 hardware, software, and support services—training, profiling, installation, and maintenance.

OCLC has also announced its new LS/2000 Micro Series of local library systems that are designed for small libraries, specialized subcollections, or libraries needing "starter systems" with upward compatibility.

Designed for those libraries that need more than a single-purpose microcomputer-based system but do not need all of the power provided by a large minicomputer system, the Micro Series is customized to individual library needs. It features Winchester disk technology in conjunction with either a Data General S/120 or DC-compatible PDP Series computer. The hardware operates in a standard office environment. Cost-saving standard profiles are available for medical, legal, corporate, academic, and public libraries. Depending on options selected, a system can support up to thirty terminals.

The LS/2000 Micro Series currently offers an online public access catalog, circulation control, bibliographic and authority file maintenance, with serials control available in 1985.

For more information, call Bob Kington, marketing and sales manager, OCLC Local Systems Division, at (614) 764-6000. ■■

Carlyle Introduces Computers

Carlyle Systems has announced a new line of computers designed especially for libraries.

The computers are used in stand-alone and distributed systems now being installed at a number of sites around the country. Earlier the company had offered a timesharing service, with terminals attached by telephone connections to large mainframe computers. This service is still available, but the new systems may be more advantageous to many libraries.

The computers are based on standard Intel boards and chips similar to those used in the IBM PC and other popular microcomputers. The larger models, however, use much more powerful microprocessors, equivalent in processing speed to large minicomputers.

The line is actually composed of several different types of computers. One type, called a remote processor, is designed to be installed at each location where there are staff or public access terminals. It handles data requests and communications traffic from terminals and contains sufficient memory to hold all error messages, help messages, and screen prompts. Another type, called a database machine, performs all database management functions, including the storage and retrieval of bibliographic records. In a distributed system, a third type, called an interface processor, connects all of the remote processors with one or more database machines and keeps track of all activity on the system.

For smaller installations, the functions of the interface processor and the database machine are combined in a single machine called a host processor, to which the remote processors are attached. For even smaller installations with only one physical location, all functions are combined in a single machine known as a user processor, and the terminals are connected directly to it. The ability to combine various types and sizes of processors in building-block fashion means that Carlyle's systems can be configured for libraries of any size and type.

Carlyle Systems is located at 2930 San Pablo Ave., Berkeley, CA 94702, or call (415) 843-3538. ■■

First OCLC M300 Workstation Installed at Hillsdale College

OCLC has announced the first installation of an OCLC M300 Workstation at the Mossey Learning Resources Center, Hillsdale College, Hillsdale, Michigan, on March 21, 1984.

The M300 Workstation, an IBM Personal Computer to which OCLC has added special hardware and software, serves libraries as an online terminal for access to the OCLC central computer system and also performs a wide range of stand-alone functions as a multipurpose microcomputer.

Hillsdale College, an OCLC participant through the Michigan Library Consortium since 1979, is a private liberal arts college with more than one thousand students and sixty-five faculty members.

OCLC plans to install approximately twelve hundred M300 Workstations to the OCLC network system this calendar year. There are presently 5,512 dedicated terminals online to OCLC, with approximately 850 authorizations for dial access to the system in addition.

In June 1983 OCLC announced that IBM would manufacture the OCLC M300 Workstation (originally called the Model 300 terminal)—an IBM PC with custom modifications made by OCLC, including OCLC's telecommunications protocol, a terminal-chaining capability, a special keyboard, and the ALA character set.

Manufacturing and distribution schedules called for M300 Workstation installations to begin in the first quarter of 1984, and the first M300 Workstation was activated on March 21, 1984, at Hillsdale College. ■■

Bob Jones Signs with Burroughs and SOLINET

Bob Jones University has announced separate agreements with the Southeastern Library Network (SOLINET) and the Burroughs Corporation to acquire services and software to implement a library automation system. The system will incorporate the LAMBDA online catalog service provided via the SOLINET online system in Atlanta, Georgia, and the Burroughs Circulation System, which will run on the university's computer in Greenville, South Carolina. The two systems will be linked via data communications through a program module that will allow full access to LAMBDA and also allow inquiry into the local circulation database to determine item status and availability.

Bob Jones University, a Christian liberal arts school in Greenville, South Carolina, maintains a 130,000-title collection in its J. S. Mack Library. The library will begin a retrospective conversion project in June 1984 on the SOLINET system. As holdings are converted, they will be left on the LAMBDA database, which will serve as the library's online catalog. In July, the library will become a SOLINET member as part of a local consortium.

The university will also install a circulation system provided by the Burroughs Corporation Software Products and Services Division in Atlanta, Georgia. The Burroughs Circulation System is to be installed on the University's Burroughs B6800 computer. As a part of that contract, Burroughs has agreed to implement a remote interface between the university's system and the SOLINET system. This interface is to be installed in late August 1984. ■■

Australian National University Awards Library Contract to AWA Computers

Australian National University (ANU) has contracted with AWA Computers for the supply of the URICA Library software system and network. The final system configuration is expected to consist of an AWA Sequel computer with 1,000 megabytes of disk memory and eighty terminals.

This decision by ANU follows a detailed evaluation of Australian and overseas library systems by senior library staff, computer services staff, and academics representing the university's Computing Policy Committee. The final short list consisted of two North American systems and the AWA URICA system. The AWA URICA software is largely Australian-developed software and will run on the Sequel computer produced by the U.S. firm Microdata, a subsidiary of McDonnell Douglas.

ANU sought suitable software before seeking quotations and determining its hardware choice. ANU's preference was for integrated hardware and software systems specialized for database management.

ANU will use the URICA software to improve acquisitions, cataloging, serials, and circulation in the eleven libraries on campus. It is anticipated that the system acquired by ANU will be operational by the second half of 1984.

For further information, contact Colin Steele, Australian National University, P.O. Box 4, Canberra ACT, 2600. ■■

Geac to Automate Bibliotheque Nationale

Geac Computer Corporation of Toronto has signed agreements with Sema, a French systems house, to automate the catalog of the Bibliotheque Nationale. This contract, valued at approximately $2 million (Canadian), spread over two and a half years, is Geac's first library automation sale in France.

The catalog, to be named the National Bibliotheque Database Management System, will be specifically written by Geac in conjunction with Sema and the Bibliotheque Nationale. When complete, the system will provide an online service throughout the Bibliotheque and, using a packet-switching network, will enable any other library within France to access its database.

In the initial phase of the project, now under way, Geac is installing 105 terminals and two Geac System 8000 computers.

Ongoing development and system support will be provided by Sema and Geac personnel in Paris. ■■

Data Phase Benchmark Test

Data Phase Corporation and Orange County Public Library have announced successful results of a performance benchmark test for an automated circulation control system. Operators performed circulation and technical services functions during the one-hour peak test using Data Phase's Automated Library Information System (ALIS II/E) software on a Tandem NonStop II computer. The same computer system is being installed by Data Phase in the Chicago Public Library.

The Orange County test, performed systemwide on one hundred terminals at twenty-six branch locations, measured transaction loads (e.g., borrowing and returning books) as projected from the current circulation of 7,150,000 to a five-year estimated circulation of 10.6 million. A peak hour is calculated by taking the current circulation and projecting the busiest hour if circulation were 10.6 million. The transaction level achieved during the test represents close to an 18 percent increase over the circulation levels originally projected, or total annual circulation of 12.5 million.

Conversion of all library locations with 1.3 million volumes to full online operations is anticipated by late spring.

Data Phase systems are currently operating in 697 libraries throughout the United States. Besides the Orange County Public Library, current users include New Orleans Public Library, Houston Area Library System, and several major universities. ■■

EBSCO Announces New Software Package

EBSCO Industries recently announced the availability of Routing, a software package for use on IBM Personal Computers. The first software of its kind, Routing allows greater control over the circulation of periodicals within an organization, the people they are routed to, and the flexibility of printing route slips.

Routing performs eleven functions, including adding titles and readers, changing title and reader information, listing titles and readers by location and by searchable identifier, and the printing of routing slips.

Routing is currently available on a five-and-one-fourth-inch diskette and will operate on an IBM PC or IBM PC XT. The Routing package will soon be available for use on other personal computers.

For a limited time only, EBSCO is offering Routing at the special introductory price of $435. For more information on Routing, contact Joe K. Weed, EBSCO Industries, Inc., P. O. Box 1943, Birmingham, AL 35201. ■■

INFO/DOC Services Now Available through UTLAS

Users of UTLAS' automated library facilities are now able to transmit orders online to INFO/DOC in Washington, D.C., for the rapid delivery of a variety of publications. This new service is the result of a recent agreement between the two organizations.

INFO/DOC is one of the oldest document delivery services in the U.S. It furnishes primarily U.S. government publications, documents, and reports, Freedom of Information items, patent materials for the U.S. and more than sixty other countries, and the literature of numerous professional societies. Suppliers include the U.S. Government Printing Office, Supply and Services Canada, National Technical Information Service (NTIS), the U.S. Geographical Survey, and others.

Users of the UTLAS online ordering system, Accord, can create order records, interfile these records with their permanent holdings, and transmit purchase orders, claims, and cancellations electronically to INFO/DOC. Other users can place orders via UTLAS' electronic mailbox facility. Orders are then filled, and the documents are dispatched on a rush or regular basis (i.e., five to ten days) at original prices, plus a small handling fee.

"Libraries taking advantage of this facility will find that they can greatly improve their service to patrons in the difficult areas of government documents and other specialized publications," stated INFO/DOC President Michael Saboe. ■■

Recent Publications

Bibliographic citations were produced with the guidance of María Clark, Yale University Library, New Haven, Connecticut, in accordance with the American National Standards for Bibliographic References. *New York: American National Standards Institute, 1977. 92p.* (American National Standards on Library Work and Documentation; *ANSI Z39.29-1977).*

Reviews

Matthews, Joseph R., ed. ***A reader on choosing an automated library system.*** Chicago: American Library Association; 1983. 340p. ISBN: 0-8389-0383-5, softcover, $35.

This book is intended to complement Matthews' earlier book *Choosing an Automated Library System: A Planning Guide* (Chicago: American Library Association, 1980). While the new volume provides a welcome second chance to experience informative and thoughtful articles from some of the most familiar and respected names in library automation—Fasana, Gorman, Hegarty, Malinconico, etc.—the book suffers (as do most technology anthologies) from the dated nature of some of the articles. Since both books are intended for small- and medium-sized libraries with plans for automation, some space should have been found for the nontechnical but often critical problems related to governance, time-sharing, and clustering. Matthews has included articles covering important topics often overlooked, such as contracts (Hegarty) and site preparation (Parkhurst). Alas, the book includes a number of stale articles on automation's impact on specific library functions. The omission of any discussion of serials control (Faxon LINX and EBSCONET, for two) is a flaw, as is the omission of new elements (mainly microcomputers) in all automation applications. For example, OCLC and downloading/copyright issues, jobber approaches to acquisitions (Blackwell, Baker and Taylor, and Brodart), and microcomputer efforts such as INNOVACQ are passed over.

Telecommunications and the microcomputer will likely play a dominant role in library automation systems from now on. The change in AT & T will fundamentally alter the way almost all libraries approach resource sharing on the local as well as the national level. The thought of using leased lines over many miles for local systems is simply not economically feasible in the deregulated environment. Adiletta's "primer" on data communications has already become a preprimer, yet it is the only article to cover telecommunications. Despite *Time* having declared the computer "Man of the Year," with specific emphasis on the personal and home computer, this book carries one article on microcomputers, and it is dated 1980, when the IBM Personal Computer had not even been introduced!

The planning and legal articles do not suffer as much from the time delay, but with so many articles focusing on technical elements that have been superseded, the audience for this book is limited. Automation managers and library school students will find the articles valuable from a historical perspective and will perhaps recognize where later developments have changed the problems and/or the solutions.

Two additional cavils: although the new volume is comparable on a per-page cost basis to the first volume, $35 seems excessive to small- or medium-sized libraries; secondly, in an era of word processors and spelling checkers, I stopped counting after the sixth typo. Recommended for comprehensive collections only.—*Robert W. White, Morris County Free Library, Whippany, New Jersey.* ■■

Gellatly, Peter, ed. ***Beyond "1984": the future of library technical services.*** New

York: Haworth; 1983. 265p. "Also published as Volume 1, Numbers 1/2 of the journal *Technical Services Quarterly.*" ISBN: 0-86656-275-3, hardcover, $24.95.

In order to predict the future of library technical services one must address a variety of questions. Will libraries in the year 2000 be museums, electronic information centers, further evolved versions of traditional libraries, centralized, or decentralized and electronically distributed? How will information be generated, stored, manipulated, and delivered? What will be the forms and functions of catalogs and other access tools? Will cataloging come to resemble indexing more closely? Is the book dying and the journal going online? Will technical and public services merge or become more specialized? Will there be more or fewer paraprofessional and clerical workers as automation becomes more specialized? Will there be more or fewer paraprofessional and clerical workers as automation becomes more pervasive? Will librarians assume more management responsibilities or will there be a greater division of professional and management roles? Will libraries become more humanistic "high touch" institutions or sleek, data-oriented "libinfocenters?" What social and economic trends will affect libraries and, conversely, how might libraries affect social and economic questions?

Nearly forty commentators, some well established, others relative newcomers, consider these and many other questions in *Beyond "1984."* Given the complexity of the issues and the inherent difficulty of forecasting, it is not surprising that no consensus emerges here. Whatever your position on such questions may be, you will find some grist for your mill in this volume and some ideas that may suggest that you reevaluate your thinking. The only relative certainties seem to be that library automation is here to stay, libraries and technical services will change—perhaps drastically—and the role of technical services librarians and staff will also change.

Although this collection is free of both high-tech hacker messiahs and querulous Luddites, the contributions do seem to reflect a continuum of attitudes. On one hand, Oboler warns of the dangers of the automated tail wagging the bibliographic dog, while Katz bemoans the computer's "little green letters [that] look like a procession of drunks in a heady wedding ceremony," perhaps recovering after participating in our society's pointless "100-yard intellectual dash." Hisatsune has a very different concern: "Second Wave thinkers among us . . . are bewildered by the innovative thoughts being expressed by a few bold Third Wave thinkers. . . . Our profession needs to break away from those traditional modes of thinking that inhibit new concepts."

One might guess from the title that issues of access to information, government and commercial restrictions, evaluations of sources, and the social impact of information would be central themes. Grams asks "Who shall benefit? Who shall suffer? The access to information by all citizens at a reasonable cost, and the right of an individual to privacy . . . may be compromised hereafter through the action of the marketplace alone." Berman points out three social pitfalls of library automation, and Gorman emphasizes the humanistic and cultural aspects of libraries. Several authors preface their remarks with warnings of nuclear annihilation, but none suggests that librarians might have some role or responsibility in preventing such a gloomy scenario.

The majority of essays, however, focus on technical and managerial questions. Hence the collection fully lives up to its subtitle, *The Future of Library Technical Services*, but falls somewhat short of the titilating title *Beyond "1984."* Given the power of socioeconomic forces to change libraries, one can imagine an entire volume devoted to external issues alone. This observation does not imply that the articles here are not interesting, relevant, diverse, and well written—for the most part they are—but simply that the scope of the volume is somewhat more limited than the title suggests.

Beyond "1984" was originally published as the first two issues of *Technical Services Quarterly*. It seems ironic that we are witnessing a proliferation of new library journals at a time when serials costs and the electronic publication of serials are impor-

tant concerns. Is this indicative of the relative health of the paper periodical or a last hurrah phenomenon? In any event, Haworth Press has been a major instigator of new library serials, not all of which are worth the paper used to print them or the electronic impulses to send them. Happily, *TSQ*, under the steady editorial hand of Peter Gellatly, does appear to have a future beyond 1984, and medium to large academic, public, and special libraries should consider a subscription.

Although few may wish to engage in an in-depth, cover-to-cover consideration of this book, many will benefit from a more selective reading. I know I'm looking forward to contemplating and discussing videodiscs, electronic document delivery, etc., while I'm up on the third floor shifting books between semesters.—*James R. Dwyer, Northern Arizona University, Flagstaff.* ■■

Nolan, Jeanne M., ed. ***Micro software evaluations.*** Torrance, Calif.: Nolan Information Management Services; 1984. 176p. ISBN: 0-915995-03-4, $45.

Nolan, Jeanne M., ed. ***Micro software report, library edition, volume II (July 1982–July 1983).*** Torrance, Calif.: Nolan Information Management Services; 1984. 157p. $49.95.

Micro Software Evaluations and *Micro Software Report* should be added to the holdings of libraries interested in the establishment of a comprehensive reference collection pertaining to currently available microcomputer software written for library applications. Both volumes succeed in the attempt to gather together a significant and useful body of information concerning such software.

Micro Software Report is a listing of microcomputer software written for library applications or directly useful for such applications. This compilation was made from citations that appeared in the literature between July 1982 and July 1983. It is a comprehensive primary reference document that will assist librarians in locating sources of relevant software programs relating to a wide range of needs. Each program listed in the volume carries a brief description, production source, and complete order information. Additionally, each entry lists a review citation for further information. Included as well are a subject index and an equipment index.

Micro Software Evaluations is a companion volume that provides in-depth reviews of "leading programs." Again, all software evaluated has direct library applications. Those applications include catalog card preparation and production, online catalog development, acquisitions, electronic mail, circulation, periodicals management, etc. A brief word-processing section is also included.

The reviewer of each program, selected at random, is also a current user of that program in a library environment. Reviews are generally jargon free, and applications of the software for specific tasks can be clearly understood.

The volume includes a set of micro software selection guidelines for librarians planning software purchases. These guidelines are well conceived and thorough. Selection guidelines are specifically written for software in the areas of periodicals management, catalog card preparation, circulation, the online catalog, acquisitions, and database management. In addition, there is a useful software matrix, which gives the reader an immediate overview of programs evaluated in the volume with relevant statistics for each program at ready reference. Selection guidelines are not included for word processing and would have been helpful, but even without them this section will save a great deal of time for librarians considering a software purchase. The volume also includes an evaluator index, institutional user index, and a subject index.

Each program evaluation includes the producer, the system requirements, and the setting in which the program is being applied. Wherever possible, the cost of the program is listed. Many reviews include "screen" examples of software program applications. These "screens" help the reader more easily understand the review narrative.

Reviewers come from all types and sizes of libraries: elementary/secondary, community college, four-year college and university, public and private agency libraries, etc. The names, addresses, and phone numbers of most reviewers are listed.

One of the shortcomings of this volume is the random selection of current program users as reviewers. Whereas most reviews are well written and objective, a few tend to be biased. Such reviews read like testimonials to the wisdom of the purchase of the program under review rather than an evaluation of the strengths and weaknesses of the program itself. A list of other users for each program reviewed would assist readers in gaining an objective insight into the application of that program in their libraries.

Some software packages evaluated are rather high powered and do not normally fit in the "micro software" category. For example, included in the evaluations are the CLSI Circulation System (cost $188,000 for hardware, software, etc.) and the Total Library System (requiring 1024K Hewlett Packard computers to operate). These reviews are well written and informative, but are not normally associated with most stand-alone floppy-disk-based microcomputer software.

Several well-written and objective reviews submitted are worthy of mention. They include reviews of the database management program dBASE II (produced by Ashton-Tate), written by John Jolly, Glendora (Calif.) Public Library; the GS 500 Acquisitions System (produced by Gaylord Bros., Inc.), by Janet Andrews and Wendy K. Robuck, Winter Park (Fla.) Public Library; the GS 100 Circulation System, by Kay Walsh, Lisle (Ill.) Public Library; the Library Circulation System (produced by Winnebago Software Co.), by Mary E. Saphner, Irving Pertsch Elementary School, Onalaska, Wis.; and the Personal Bibliographic System (produced by Personal Bibliographic Software), by Wendi Pohs, *Mathematical Index and Abstracting Journal*, Ann Arbor, Mich. Each of these reviews illustrates microcomputer program software that is being effectively used in a library environment and that could reasonably be applied in most small- to medium-sized libraries, both public and private.

The review of MicroCheck (produced by Brodart, Library Automation Division), a program designed to assist in the production of computer output microfiche (COM) catalog updates as an alternative to manual systems using standard typewriters with optical character recognition (OCR), is an example of a review of a program with technical specificity but with potential applications in many libraries. Other examples of such programs included for review are the Overdue Collector and the Overdue Writer (produced by the Library Software Company), Checkmate (a periodicals management program produced by CLASS), and Catalog Card and Label Writer (produced by Wehner Education Software).

Micro Software Report and *Micro Software Evaluations* are important additions to the body of reference literature available to librarians considering the purchase of micro software for professional applications. The programs listed and reviewed represent the most recent examples of library software available for purchase. Both volumes are distinct. *Micro Software Report* provides one of the few compilations of library-specific microcomputer software in print. *Micro Software Evaluations* attempts to take the "leading programs" written for library applications and provide an in-depth, user-referenced review. With the weaknesses noted, both volumes are successful and their acquisition is recommended.—*George Bynon, University of Oregon, Eugene.* ■■

Rice, James. *Introduction to library automation.* Littleton, Colo.: Libraries Unlimited; 1984. 209p. (Library science text series). ISBN: 0-87287-413-3, hardcover, $28.50 in U.S. and $34 elsewhere; 0-87287-433-8, softcover, $18.50 in U.S. and $22 elsewhere.

In his preface, the author, James Rice, indicates a threefold objective for his book. It is intended to serve students as a textbook for courses in library automation, to serve practicing librarians studying independently, and to inform the administrators or planners who design and select automated library systems. The "overall goal of the text is to be a readable, brief introduction to library automation" without "technical verbiage or engineering jargon." These are certainly laudable goals. They can be achieved through a careful balance of practice and theory, with practical advice and theoretical explanations selectively interwoven throughout the text. Rice, however, does not achieve the needed balance in this book, and in trying to serve the three masters of student, practitioner, and administrator he serves none well.

The eighteen chapters are divided into four units: Introduction to Automation; An Overview of Library Automation; Determining Needs and Making Decisions; and Implementing the System. The first unit, Introduction to Automation, is the weakest of the book. In less than 50 brief pages the author attempts to cover the topics of computer history, basic computer design, telecommunications, word processing, machine-readable conversion, file structures, and input, output, and storage devices. The author seeks to provide an informative summary, but in trying to cover so much material in such a small amount of space, he presents not a helpful summary of data processing basics but a catalog of concepts with explanations that raise more questions than they answer and frequently omit important considerations. The complete explanation of punched paper tape, for example, takes two very brief paragraphs with no indication of any of its disadvantages or datedness.

Indeed this criticism of the first unit can be extended to the entire book. If one discounts glossary, bibliography, and index, there are fewer than 170 pages of text. In this space Rice attempts to provide a review of all the important issues and concepts related to library automation, systems analysis and design, system implementation, and computing fundamentals. The balance between detail and summary so critical in such an attempt is not achieved. In some chapters the detail is excessive (e.g., chapter 18 advises frequent cleaning of CRT screens, keyboard, and vents for cooling fans); in others the topic desperately cries out for more attention (e.g., the single page given to automation of serials in chapter 11 or the example of punched paper tape given above).

Illustrations would have helped the book by explaining in diagrams what could not be detailed in words. However, to introduce basic computing devices and concepts the first unit uses just five illustrations. Even these do not clarify or fill in for the text (one is a list of EBCDIC symbols, for example); others are too simple to be of any substantive help (two of the five illustrations are versions of the obligatory input, process, output box diagrams). Again, the criticism can be extended to the whole book. It has a total of just seventeen illustrations and over half of these are example flowcharts.

The book is certainly not without value or usefulness, however. The first unit would perhaps be acceptable for textbook purposes where the lack of depth could be supplemented with lectures. Most of the book's value, though, comes in the later chapters in which specific advice is given. An example is chapter 17, "Conversion of Library Records to Machine-Readable Form." Here the author lists steps to take for successful conversion, discusses the role of large machine-readable databases (LC MARC, REMARC, OCLC, UTLAS, etc.), and briefly describes some examples of conversion projects. Another strength of this chapter is that it is easily the most documented of the book, with forty references. These references serve as a good source of information where the detail in the text is insufficient for the interested reader. The book also has a good, selective topical bibliography in back.

Note, however, that these later chapters are composed of a number of timetables and lists of steps to take. They would be useful for those with a good background in automation, for example, library analysts and administrators desiring a quick overview and guide to action. These later chapters would be much less helpful to either the student, who needs a less specific approach, or the librarian unfamiliar with the computer, who needs a more detailed analysis of the issues and options involved. It is interesting to quickly compare the successful approaches of some other authors to the topic of library automation. Saffady's recent *Introduction to Automation for Librarians* has similar objectives and intended audiences similar to this book's but

takes about twice the number of pages to achieve them. Other briefer texts succeeded by having more selective goals and focusing on a particular aspect of automation. Matthews' *Choosing an Automated Library System* is a good example.

If a library is considering automation in the near future and wants every piece of advice it can find in the literature, Rice will provide some help. A better selection for a comprehensive approach, however, is Saffady. If information on a particular aspect of automation is needed, such as selection, then a book that focuses on the topic is generally a better choice.—*David Carlson, University of Rhode Island, Kingston.* ■■

Rubin, Michael Rogers. *Information economics and policy in the United States.* Littleton, Colo.: Libraries Unlimited; 1983. 340p. ISBN: 0-87287-378-1, hardcover, $35 in U.S. and $42 elsewhere.

Some librarians do not wish to get involved in issues related to the "right to know." But for those who do, this comprehensive collection of documents on information-related issues provides an excellent resource for understanding these topics. Although the book virtually ignores the role of libraries, it is loaded with essential information useful to librarians grappling with this highly complex subject.

The author, known to librarians for his work on the government report *The Information Economy* (with Marc Porat), is an attorney who has served as Special Assistant for Information Policy to the Deputy Assistant Secretary of Commerce for Science and Technology. Many chapters in the book reflect his experience in this position at the Department of Commerce during the Carter Administration, when information policy received significant attention. Unfortunately, most of the documents in the book predate the current administration, which has reversed so many of the efforts set forth in the late 1970s, redefining the right to know in a far more restrictive manner. Hence, the book is better for historical perspective than current awareness. And because the author strongly believes that government involvement in information policy is essential, his perspective hardly reflects the contemporary political approach (or lack thereof) to the so-called information dilemma.

The keystone of the book is the emergence of new technologies and their effects on labor, economics, and privacy. After presenting a succinct review of recent developments in information technologies and an assessment of their impact on the marketplace, Rubin considers policy implications by examining the role of the federal government in dealing with the effects of information innovations on society.

Each chapter begins with a short essay by the author that introduces the topic, highlights the policy-related issues, and is followed by key relevant documents. For example, in the section on the economics of information, Rubin incorporates an excerpt on the definition and measurement of the information economy by Marc Porat, followed by a previously unpublished translation of Simon Nora and Alain Minc's "Telematics and New Growth," from the 1978 *Report on the Computerization of Society* to the President of France. This broad array of approaches enhances the value and credibility of the book—a book that is no mere stab at this highly complicated, barely explored topic.

Among the other subjects covered in the text is transborder data flows. In this chapter, Rubin looks closely at several European approaches to data protection, comparing laws in West Germany, France, and Sweden. He then points out the serious consequences of erecting barriers to the free flow of information, either through restrictive international laws or by overlapping, contradictory national laws. American responses to these walls that are being built around nations have been fragmented and piecemeal because of little awareness of the problem and no national policy to meet it.

Another well-covered topic is government participation in the marketplace, a subject hotly debated in recent years between librarians and representatives of the information industry. Rubin provides a philosophical counterpoint to the arguments often presented. His supporting documents are useful to understanding the various positions taken on this issue. They are five years old, however, and therefore practically obsolete in today's political environment.

Rubin's last two chapters present topics of particular interest to librarians: personal privacy and the need for a national information policy. The dimensions of the privacy issue range from intrusions into personal matters to preventing the misuse of personal information an individual has disclosed voluntarily. Rubin traces the concept of privacy from the Bill of Rights through more recent legal doctrines and then presents proposals based on recommendations of the Privacy Protection Study Commission for regulating possessors of personal information. Unfortunately, the commission's three concurrent objectives—to minimize intrusiveness, to maximize fairness, and to create legitimate, enforceable expectations of confidentiality—have resulted in the passage of too little legislation to assure the preservation of privacy in relation to far-reaching social and technological changes.

The diversity of issues presented in this book, as well as many that are ignored, constitutes the spectrum of information issues for which policy responsibility is scattered among numerous federal agencies. This results in a balkanization of policy. One solution proposed by Congressman George Brown in his "Information Science and Technology Act of 1981" (which was closely monitored by ALA) provided a mechanism to develop an integrated approach toward information access and delivery. Many were in agreement on the need for policy definition but were less supportive of his recommendation to develop a body to do so within the executive branch.

The book ends abruptly with testimony related to the Brown bill. This may be an appropriate means of closing a chapter on our government's attempts to deal with the implications of the information age, since it is doubtful that further federal action will be taken any time soon. What that means for the flow of information and the role of libraries is anyone's guess. But it clearly indicates that many of the dilemmas facing our information providers will probably not be resolved by a central government initiative in the near future. Hence, it may remain up to many independent uncoordinated forces to wrestle with these issues. Recently, some librarians have indicated a reluctance to participate in information policy-making. This could drastically affect the supply and demand of the very commodity librarians so conscientiously promote—information.

In sum, although this book needs to be supplemented by more up-to-date reports on the state of information policy in this country, it is an invaluable tool for understanding many issues confronting an information society. For those wishing to pursue this topic further, the footnotes and other references provide an excellent array of materials worth adding to any collection on this subject.—*Nancy C. Kranich, New York University Libraries, New York.* ■■

Tolle, John E. ***Public access terminals: determining quantity requirements.*** Dublin, Ohio: OCLC; 1984. 161p. (OCLC library, information, and computer science series; 3). ISBN: 0-933418-51-5, softcover.

For anyone involved in the specification of an online catalog, all help is clearly welcome. The intent of this book is to give guidance in the area of determining terminal requirements. Based on a project at the Ohio State University Libraries, it details a research methodology and demonstrates its use. It is not, however, for the mathematically faint of heart.

The book is divided into five chapters. The first provides a brief introduction and problem statement. As the author indicates, this chapter is useful reading for "anyone interested in or responsible for an online system." The fifth chapter outlines and summarizes the methodology for conducting a terminal requirements study. Contrary to the implication that this chapter stands on its own, frequent references are made to the more technical chapters in the book, and much of the terminology requires familiarity with those chapters.

Chapter 2 details the mechanics of data collection in a terminal requirements study. The methodology is presented quite clearly and could be applied to other types of studies as well.

Chapter 3 presents the development of analytical mathematical models through the application of probability theory. The author indicates that the chapter "may be skipped or skimmed by those not mathematically oriented," but one cannot use the guidelines in Chapter 5 without some familiarity with the concepts presented here. In Chapter 4, the author applies the mathematical models to the collected data to predict terminal requirements. This step is the culmination of the process, but it is still not easy to follow.

This book proposes to give guidance in answering the question of "how many terminals are enough to meet user needs." Unfortunately, the guidance is nestled in a mass of equations, graphs, and tables. Anyone looking for a quick and easy answer will not likely find it here.—*Alan E. Hagyard, Yale University Library, New Haven, Connecticut.* ■■

Other Recent Receipts

Listed here are books and other publications of potential interest to members of LITA, received for review. Some of these materials may be reviewed in later issues of ITAL.

Aschner, Katherine, ed. *Taking control of your office records: a manager's guide.* White Plains, N.Y.: Knowledge Industry; 1983. 264p. (Information and communications management guides). ISBN: 0-86729-057-9, hardcover, $32.95; 0-86729-058-7, spiral-bound, $22.95.

Beaumont, David; Summerfield, Anne; Wright, Julie. *Working with RT-11.* Bedford, Mass.: Digital Pr.; 1983. 216p. (RT-11 technical user's series). ISBN: 0-932376-31-2, softcover, $19.

Boss, Richard W. *The library manager's guide to automation.* 2d ed. White Plains, N.Y.: Knowledge Industry; 1983. 169p. (Professional librarian series). ISBN: 0-86729-052-8, hardcover, $36.50; 0-86729-051-X, softcover, $27.50.

Computer publishers and publications: an international directory and yearbook. 1984 ed. New Rochelle, N.Y.; New York; London: Communications Trends; 1984. 383p. ISBN: 0-8103-0540-2, hardcover, $90. "North American library edition distributed exclusively by Gale Research Company."

Databases for books: their uses for selling, acquiring and cataloguing. Proceedings of the MARC Users' Group Seminar, 3 May 1983. London: Library Assn.; 1984. 62p. ISBN: 0-85365-656-8, softcover, $15. Available in U.S. from Oryx, 2214 North Central at Encanto, Phoenix, AZ 85004.

Directory of public domain (and user-supported) software for the IBM Personal Computer. Santa Clara, Calif.: PC Software Interest Group; 1984. 116p. ISBN: 0-915835-01-0, softcover, $4.95. Available from: PC-SIG, 1556 Halford Ave., Suite #130, Santa Clara, CA 95051.

Helal, Ahmed; Weiss, Joachim W., eds. *New trends in electronic publishing and electronic libraries.* "Essen Symposium, 29 August–31 August 1983." Essen: Gesamthochschulbibliothek Essen; 1984. 196p. (Veröffentlichungen der Gesamthochschulbibliothek Essen; 6). ISBN: 3-922602-07-X, softcover.

Heynen, Jeffrey, comp. *Microform sets in U.S. and Canadian libraries: report of a survey on the bibliographic control of microform sets conducted by the Association of Research Libraries Microform Project.* Washington, D.C.: ARL; 1984. 118p. ISBN: 0-918006-08-2.

IBM 4361/4381 processors in-depth report. "Report to assist users in evaluating the full impact of the 4361/4381 announcement." Medford, N.Y.: Applied Management Services; 1984. 68p. Softcover. Available from: AMS, Box 350, Medford, NY 11763.

Inside IBM. Medford, N.Y.: Applied Management Services; 1983. 148p. Softcover, $120. Available from: AMS, Box 350. Medford, NY 11763.

International MARC Project. *International guide to MARC databases and services.* 1st ed. Frankfurt, Germany: Deutsche Bibliothek; 1984. 136p. ISBN: 3-922051-08-1, softcover, $18. Available from: IFLA International MARC Programme, Deutsche Bibliothek, Zeppelinallee 4-8, 6000 Frankfurt am Main 1, Federal Republic of Germany.

Kenney, Brigitte, ed. *Cable for information delivery: a guide for librarians, educators and cable professionals.* White Plains, N.Y.: Knowledge Industry; 1984. 172p. ISBN: 0-86729-056-0, hardcover, $34.50; 0-86729-055-2, softcover, $27.50.

Linking: today's libraries, tomorrow's technologies. Ottawa: National Library of Canada; 1984. 63, 72p. (Canadian network papers; 7). Text in English and French. ISBN: 0-662-52889-1, softcover.

Maciuszko, Kathleen L. *OCLC: a decade of development, 1967–1977.* Littleton, Colo.: Libraries Unlimited; 1984. 376p. ISBN: 0-87287-407-9, hardcover, $45 in U.S. and $54 elsewhere.

Miller, Inabeth. *Microcomputers in school library media centers.* New York, N.Y.: Neal-Schuman; 1984. 165p. ISBN: 0-918212-51-0, softcover, $19.95.

Miller, Jerome K. *Using copyrighted videocassettes in classrooms and libraries.* Champaign, Ill.: Copyright Information Services; 1984. 91p. (Copyright information bulletin; 1). ISBN: 0-914143-00-X, softcover, $10.

Neufeld, M. Lynne; Cornog, Martha; Sperr, Inez L. eds. *Abstracting and indexing services in perspective: Miles Conrad Memorial Lectures 1969–1983.* "Commemorating the Twenty-fifth Anniversary of the National Federation of Abstracting and Information Services." Arlington, Va.: Information Resources Pr.; 1983. 300p. ISBN: 0-87815-043-9, hardcover, $27.50 plus $2.10 for postage and handling.

Oboler, Eli M. *To free the mind: libraries, technology, and intellectual freedom.* Littleton, Colo.: Libraries Unlimited; 1983. ISBN: 0-87287-325-0, hardcover, $15 in U.S. and $18 elsewhere.

Troutner, Joanne. *The media specialist, the microcomputer, and the curriculum.* Littleton, Colo.: Libraries Unlimited; 1983. 181p. ISBN: 0-87287-367-6, hardcover, $19.50 in U.S. and $23.50 elsewhere.

Woods, Lawrence A.; Pope, Nolan F. *The librarian's guide to microcomputer technology and applications.* White Plains, N.Y.: Knowledge Industry; 1983. "Published for American Society for Information Science." 207p. ISBN: 0-86729-045-5, hardcover, $34.50; 0-86729-044-7, softcover, $27.50. ■■

INDEX TO ADVERTISERS