A Role for the Minicomputer in Library Education

This paper discusses how one library school, the Faculty of Library Science at the University of Toronto, uses a minicomputer. The pleasures and problems we have experienced with our mini relate to the environment of the school and its educational objectives. They are not necessarily generalizable, but they may provide some insights into the potential of minicomputers. At the outset we should emphasize the newness of our system. We are feeling our way; undoubtedly we are making some mistakes, but we are learning a great deal in the process and are very optimistic about the future. We hope to demonstrate the potential of minicomputers for library education in a way that will be useful for anyone considering a mechanized support system and concerned about the expense and commitment of a large-scale operation.

This paper should be interpreted in the context of our particular situation. Some background on our school, its program, and some description of our environment will make our remarks more meaningful. FLS is old as library schools go. Its forerunners, the Ontario Library School and the University of Toronto School of Library Science, date back to 1928. It has been accredited since 1937. By the mid-1960s it had grown to be one of the largest library schools in North America. It has a reputation for being slightly traditional, but also for turning out students who have solid grounding in the science of librarianship.
If the school is old in years, it feels, to those who are with it now, very young in a number of ways. In the course of the last few years there have been many changes, and some of these have been abrupt and dramatic.

The one-year postgraduate Bachelor of Library Science degree has been replaced by the two-year Master of Library Science as the basic degree. In the old baccalaureate program, the students felt they were on a treadmill. With everything squeezed into eight months, there was little time for in-depth discussion of theory or for research. Only the few students who returned for a further year of study to qualify in the old MLS program had much opportunity to follow areas of individual interest. The switch from a one-year degree to a two-year degree was not made primarily to stretch the time. A totally fresh approach was taken; an entirely new curricular concept was developed. The demand for trained decision-makers, systems planners, and researchers needed to be met. The new curriculum, now in its fourth year, is characterized by the identification of five core areas: the library and its community, resources and collections, organization of materials, library administration, and library research. Basic courses in these areas are taken by all students and a wide range of electives is offered. Each student is free to design the program which best suits his needs and interests. There is an apparent absence of library automation in the above list. Computers, as tools to support library operations, are dealt with in all the core areas. In addition, each student must fulfill a computer programming requirement and may opt to specialize in library automation through the electives he chooses. With this time stretching, there is provision for in-depth study. And the implications of this for an in-house computer facility are great.

The full-time teaching staff has more than quadrupled in number in the past ten years, making specialization possible for individual members of the faculty. Of the twenty-two professors on the full-time staff at present, ten hold doctorates and six others are working toward doctorates. One year ago, official government blessing was given to the new Ph.D. program which had been launched at the school. A recent count identified nineteen faculty and Ph.D. students engaged in research or about to embark on projects.

Last but not least, the school has occupied a magnificent new seven-story building for the past three years. The FLS building is attached, siamese twin fashion, to the new Robarts Library of the University of Toronto which houses the university’s research collection in the social sciences and the humanities. The computer-based cataloging project of the newly formed Ontario University Libraries Cooperative System also has its headquarters in the Robarts Library. Occupying two floors of the new FLS building is the FLS library with its specialized collection of 50,000 volumes, a collections
budget of about $38,000 and a staff of eleven. The library provides a working
model and a laboratory in which we can demonstrate special techniques and
approaches for the future librarians who are our students. This model concept
has enabled the FLS Library to gain benefits from our automation activities
which could not be justified with cost-effective arguments alone.

An integral part of our new building is a full-fledged data processing
laboratory with adequate space, atmospheric controls, provision for subfloor
wiring and an underfloor cable system throughout the building. This provision
for data processing equipment reflects the faculty’s philosophy with regard to
electronic data processing in library education. The importance of knowledge
about automation in libraries is irrefutable. It is safe to say that, with the
rapid emergence of bibliographic data bases and cooperative library systems
and with decreasing hardware costs bringing mechanization to more and more
libraries, the importance of this knowledge for librarians is on the increase.

The positions held by some of our recent graduates bear this out. A
number are now systems librarians actively engaged in planning and imple-
menting new mechanized applications in libraries. Others are working more
effectively with automated systems because they were able to acquire some
skills in this area at FLS. Some now wish they had taken more of the
computer-oriented electives. Some older graduates are returning to up-grade
their knowledge in this area through the computer electives program. Autom-
tation workshops in our continuing education program are one of the high
priorities.

Data processing was introduced to the FLS curriculum in 1966. From
the beginning, it has been in the hands of library science faculty. This has
insured a consistent policy and strong library orientation throughout. An
objective then, as now, was to give interested library school students as strong
a grounding as possible in computer technology and its library applications, to
facilitate good communication with machine people and sound decision-
making in this area.

Our first facilities took the form of a crowded, unairconditioned unit
record laboratory with keypunch, sorter, collator, document writer and ac-
counting machine, complemented with batch use of an IBM 7094 at the
University of Toronto Computer Centre. Automation courses in the baccala-
reate program were all optional and minimal—two hours per week of class
time for one term including some simpler programming.

In the first year of the new master’s degree program, all students
complete a set of computer programs—the computer requirement mentioned
earlier. This uses a ten-instruction simplified computer, simulated on the
university’s IBM 370, in batch mode. Our objectives with this assignment are
to introduce sequential step methodology in problem-solving, to convey the binary nature of machine decisions, to identify human factors in the man-machine interface, to establish standards for documentation of projects, and to awaken interest and aid the students in identifying their interest in and aptitude for specialization in this area.

Consistent with our hands-on approach are a number of on-line simulations which have been developed as teaching aids for the core courses to familiarize the uninitiated with terminal interaction and to demonstrate the capabilities and sophistication of on-line systems. In particular, we have prototypes of an on-line cataloging system and an on-line information retrieval system. Both CRTs and typewriter terminals are used for these demonstration systems. Also, through the cooperation of the National Science Library of Canada, students have an opportunity to experiment with profile construction for an SDI system.

For students who choose to specialize in the automation area in their second year, we offer a program including studies in documentation techniques, characteristics of information structures, file organization, system design, and information transfer patterns. The six courses fall into two separate but related streams. The documentation stream has a special library orientation and is aimed at the librarian who will be working alone to provide information services to the specialist. The systems stream concentrates on library housekeeping operations, MARC format, file management, and systems design, and is aimed at the librarian who will be planning and implementing large-scale applications as one of a team. All six courses blend theory with practical applications. Most include projects which require programming and other machine interaction.

For the second-year student who has identified a special library problem, the curriculum offers a research stream option which releases him from course responsibilities in order that he may pursue his specific project. Already students working in this stream in nonautomation areas have made good use of computer support services for analysis of data. To date, computers have been used in the doctoral program in a supportive way: for text editing, for statistical analysis of data, and for simulation studies.

We are also building up a first-rate research facility. For this we need a number of things. We need the climate for research, which we have. Our teaching program is research-oriented at both the master's level and the Ph.D. level. Faculty strength in the area of automation and systems has been enhanced by cross appointments from other university departments, including computer science. Staff and staff expertise in information processing and systems are being built up to support a Ph.D. major in the future. A number
of faculty are presently engaged in computer-related research and others are using support routines already mentioned. Studies are underway on cataloging practice, indexing effectiveness and thesauri, and using the bibliographic data base of the University of Toronto Library Automation System. To support this research activity, we needed a laboratory, the physical space, and controlled atmosphere for the equipment—and for the people using it. Our new building has provided this. We needed equipment which is sophisticated, up to date, and which provides access to a wide variety of facilities. Variety would insure the flexibility which is so very important in an environment supporting both teaching and research.

It was to support just this philosophy that the FLS undertook, in the period from 1966 to 1970, to develop the functional specifications of computer services for its new quarters. As the 1960s gave way to the 1970s, however, the actual implementation of these specifications differed somewhat from the initial ideas because of changes in the economic climate and in technological developments.

Between 1966 and 1968, the staff of the school—W. J. Kurmey in particular—examined methods for using the computer to the best advantage in the school’s program, and these methods were tested, using available equipment in a limited application. With the assistance of other staff on campus, preliminary ideas were developed into a set of requirements. Basically these requirements specified two types of environment: standard batch and complex interactive. Both would have to support the peculiar requirements of bibliographic information storage and retrieval processing (e.g., large data bases, large character-sets and character-string processing, as opposed to straight computational processing), in addition to the general computational functions needed for research, computer-assisted instruction, and administrative processing. It was anticipated that the interactive facility would have a large number of CRT terminals both for teaching and research.

Negotiations with the staff of UTCC and of UTLAS led to the conclusion that neither of these operations could support the projected requirements of the school. While the UTCC facility could provide the general batch services needed by the school, it was not prepared to make the additional resources available for interactive processing. The UTLAS staff, although heavily engaged in interactive processing, did not feel that it could risk the problems resulting from having students and experimental processing in an operational system.

In late 1967 the school, with the help of the UTCC staff and others, developed a proposal for a satellite computer system (see figure 1). The principal function would be to serve as an interface between CRT terminals
and the other facilities on campus. In 1967 facilities for a data processing laboratory in the new building were planned using as a model an IBM 360/30 with local disk and tape storage, a high-speed upper/lower case printer and
considerable communications equipment. In January 1969 the school received a capital appropriation from the provincial government to be used for the purchase of hardware.

In May-July 1970, with a firm date for occupation of the new building, tenders were received from several computer manufacturers: a Xerox Sigma 5, an IBM 360/30, a UNIVAC 9400, and a RCA SPECTRA 70/35. Although the systems tendered did not meet our specifications completely (or, if they did, were too expensive), the school’s committee settled on one of the proposals, accepted the compromises it entailed, and sent its recommendation to the director of the school. By September 1970 the system had been approved by the Technical Subcommittee of the University Computer Policy and Planning Committee. FLS seemed to be on the way to having a medium-sized computer operation.

By 1970, however, a number of things had changed. The most significant was in the economic climate. The school had the capital budget for the equipment, but it did not have the operating budget to support a system such as that being proposed. The university administration, in approving the purchase of the recommended system, would also have to approve an additional annual budget expenditure of between $70,000 and $100,000 for staff and maintenance. They were not prepared to do this, and in late 1970 the school was asked to review its requirements and come up with alternative solutions. In the year and a half that followed, the previous five year’s work had to be rethought and, fortunately, other changes occurred which made this easier.

The first of these changes was the decision of the new director of UTCC to provide an interactive service on a large scale. The interactive system that he proposed for the university was an IBM 360/65, separated from the general batch operations on an IBM 370/165, and dedicated to APL, CPS, high-level programming languages, and ATS, a text editing facility. During the early months of 1971, the school experimented with APL. The results showed that although APL was useful for a number of things which we wanted to do, it had serious limitations. In late 1969 and 1970 the school had experimented with CPS and found it to be far from satisfactory. The principal problems of both systems were the file and program size and, with APL in particular, the restriction on the type of terminal that could be used. It was not possible, for example, to use CRT terminals, and there was a severe limit on the size of the character-set which could be processed. APL, as implemented by IBM, was effectively limited to IBM 2741 type terminals. In July 1971, at the suggestion of the UTCC director, a minicomputer was ordered with the software to effect the code conversion needed to interface CRTs to APL. By summer 1972, this system was being used regularly for manipulation under APL of a
technical reports file. As a result of this experience, we felt that APL, inter-
faced through a mini, could be used for some of our processing.

At the same time that the mini was being introduced into the school,
there was a change in administration at UTLAS and with it a change of
direction in planning for its future. During the months of late 1971, an
agreement for shared facilities was worked out between the faculty and the
university library. Under this agreement, the faculty purchased major pieces of
hardware for use by UTLAS: a CPU, core memory and some peripherals. In
return the school was given use of the system, a Xerox Sigma 7 and a Xerox
Sigma 6, which, because of the expansion and a new operating system, could
support some of the processing the school wished to do. In addition to the
normal computer services which this timesharing facility gave the school, there
was also access to the Xerox Graphic Printer with its special characters and
type fonts.

Most important, however, the agreement has involved the school in a
system which is the center of Canada’s largest machine-based bibliographic
network. It gives the school access to over 15 million records including
LC/MARC, BNB/MARC, and Canadian/MARC. In addition, as a spinoff from
the basic agreement, the school and the library have jointly developed a
special purpose CRT for bibliographic processing.

As a result, FLS became involved with three systems: the IBM 370/165
for general batch, the IBM 360/65 for interactive APL, and the Xerox
Sigma 6/7 for other processing, both batch and interactive. This meant that
our communication problems were even greater than in the case of the earlier
system. Several types of terminals were proposed and planning was underway
to use all three systems and often the same data on all of them. By early
1972, we had gained enough experience with the capabilities of the mini-
computer and sufficient knowledge about the type of system which our
vendor, GEAC Computer Corporation, was designing that in May 1972 we
asked them to work with us in designing a system which would meet
communications and local processing requirements of the school.

Our present system (figure 2), based on the minicomputer, is func-
tionally not greatly different from the one proposed five years ago—it costs
about one-third as much and has several other advantages as well. At present,
the maintenance costs for the system are less than $10,000 per year—the staff
consists of one student working less than 10 hours per week and about
one-third of the administrative assistant’s time. One of the major reasons for
the low staff costs is that the system was designed as a turn-key system with
operator intervention only when serious errors occur or in starting or termi-
nating operations.
Fig. 2. Present Data Processing System
Besides the difference in cost, another advantage of the mini system is that of flexibility in both the hardware and the software. The school's staff had much more control over design of the software than would have been the case in a larger system. At the time of our contract, the vendor was designing an operating system (of which ours is a subset), and we had considerable input into this design. The communication software, designed under our direction, has taken advantage of many local factors which standard package software could not have done. The school also had much more control in the choice of equipment. When considering the tenders for the larger systems, we often felt "Wouldn't it be nice if we could have that printer on this CPU?" With the mini, we were able to make good, cost-performance compromises and buy equipment which we felt could best meet our needs. Thus, we have a Hewlett Packard computer with a CDC disk drive, a Mohawk Data Sciences printer and card reader and several types of terminals—all possible because it is comparatively easy to connect different equipment to the mini.

Inherent in the advantages arising from the flexibility of the system are some disadvantages from the point of view of an educational operation. First, the equipment was bought, assembled and integrated by a middle-man arrangement with the result that it is not a standard system such as we would expect our students to encounter in the field. Secondly, we do not have the stability of a system backed by a large corporation. Finally, we lack the standard language and operating procedures available with a "big name" system. It is felt, however, that these disadvantages are more than compensated for by the mini's flexibility and by the variety of facilities available to our students and faculty.

Having recounted the tortuous path by which the school arrived at its present system and some general statements about its advantages, reasonable questions at this point would be: What does the system consist of in terms of specific software and hardware? How does it work? What can it do? Functionally, the system as it stands has four major components.

OPERATING SYSTEM

The operating system is a simple one providing the minimum services needed to manage the computer resources in a multiprogram environment. These services include memory allocation, execution control and input/output control. Memory allocation is handled by the loader which brings programs into memory, checks for conflicts of usage, supplies the required information to the system task-control tables and starts execution. A subset of the loader is an executive routine for overlaying program segments and is called from the user program. Execution control
is handled by a scheduler which can put a program in any one of six states and uses a priority chain. The six possible states are execution, suspension by a higher priority task, input/output wait, input/output complete wait, executive service wait, and termination processing. Normally a program will execute unless interrupted by a higher priority task, until there is need for input or output. When an I/O request is made, execution is passed to another program. After the I/O request has been satisfied, the program enters the I/O complete wait and will continue execution when the scheduler reaches its position on the priority chain. The executive service wait occurs when the program has requested a system service module which is not presently available, e.g., a call for an overlay. Termination processing is a temporary state during which the system completes the necessary "housekeeping" needed to release a program from the system.

Input/output control is handled by an I/O subsystem which manages requests for I/O devices, calls the appropriate device drivers and processes device interrupts. The I/O device drivers for the line printer, the disk and the multiplexers are integral parts of the operating system. Drivers for other devices can be added to the system or included as part of the application program using standard linkages to the system's I/O controller for interrupt and error processing.

Supporting the system and its use is a general service subsystem which provides access to the software needed to input, test, and debug applications programs and some system utilities such as the file purge routine and file directory listing. From this subsystem the user can access an editor for creating and updating source programs, the assembler and the loader. This service subsystem operates much like a background mode on larger computers but is seen by the system as a large special program. For debugging purposes, programs are run under the control of this subsystem before being added to the inventory of regular programs. Debug aids are in the form of snapshot dumps.

Another of the support subsystems is the spooler, a routine which primarily controls output to the line printer and input from the card reader. It can, however, be used to create a temporary file when passing data from one program to another, when that is necessary and shared access is not desirable.

The languages available on the system are GEAC's version of the Hewlett Packard assembler, Fortran, Basic and OPL, a high-level language designed by GEAC with the capability of having inbedded assembler level instructions in the source program.
REMOTE COMMUNICATIONS

There are several remote communications modules. Using synchronous communication, the system has direct remote batch connections to both the IBM 370/165 and the Xerox Sigma 6/7 systems. Indirectly through the 370, special batch jobs can be run on the IBM 360/65 for retrieving and storing data in the APL file system. The remote batch facility allows jobs to be transmitted to either of the two machines, allows status inquiry to be made from a CRT terminal and, in the case of the 370, prints the output either on the school's printer or on any of the other remote printers on campus or on the UTCC printers. For the Xerox system, output can be printed either at the school or on the printer at the UTLAS site.

In the planning stage are two more remote communication functions. First, it is planned to provide the capability of intercepting output from one system and rerouting it to another, e.g., data coming back from the 370 could be intercepted and sent to the Sigma 6/7. Thus, for example, new data created under APL on which we have several data entry and edit programs could be off loaded from the APL files and sent to the Xerox system. It is also planned to provide asynchronous communication to the IBM 360/65 to replace the earlier APL interface program and asynchronous communication to the Xerox Sigma 6/7. The objective of the latter software is to allow automatic log-on and provide easy switching from one system to another without the mechanical changes that are presently required. It is felt that this will be especially useful for computer-assisted instruction and research activities where the user is not primarily interested in, or knowledgeable about, the various systems but requires the use of predefined processes such as statistical packages, simulation routines, etc.

Consideration is being given to FLS being a member of OULCS's network with the mini as one drop point on the network. This would give FLS access to the bibliographic system as a regular user. Another possible link which may be considered in the future to enhance the range of facilities would be to the information retrieval systems being established by Canada's national libraries. Should other external systems offer new dimensions, the flexibility of the software and the hardware will make it possible to communicate with them.

TERMINAL COMMUNICATIONS

At present the system's terminal communications can handle sixteen terminals or terminal-like devices using standard asynchronous protocol. The terminals are handled by two multiplexers which were designed and built by
GEAC and which have the capability of providing various speed and code configurations on each port, i.e., speeds from 110 to 9600 baud and 5-11-bit character codes. In the planning stage is a synchronous, terminal communication subsystem for the special bibliographic terminals.

FILE SUBSYSTEM

There is a very simple file subsystem which provides the necessary control information for managing source files with the associated relocatable and load module files. The source file part of the subsystem provides blocking and unblocking of the short record which would be created by the editor. The subsystem also provides the control information for user files. These files are made up of contiguous sectors of disk storage which are written or read a sector at a time by requests to the file subsystem. The user is responsible for his own data management routines within the block of disk storage assigned. The system disk, a CDC 9746, provides about 50 million bytes of storage. Since the present system with its various temporary files, source, relocatable, and load module program files, etc. requires only about 5 million bytes of storage, there is plenty of room for local applications and future expansion. Tape files can be created and read in much the same way as tape usage on other systems except that, without an operator, use of tape is limited to users who have been trained to mount and demount tapes. For the present, the tape drive is primarily to be used for back-up and special applications.

As one might guess, the design and implementation of the system which has been described has been a challenge. For the most part, difficulties stemmed from one general problem—lack of experience and available expertise. The vendor, although quite at home with the hardware and other types of applications, was not familiar with library needs or with the school's specific educational objectives. The experience of the school's staff had mainly been with big systems, yet in order to make decisions and provide the information which the vendor required they often had to deal with a level of detail which would not have been true in an off-the-shelf system. Frequently there were delays while the information needed for a decision was acquired. The other units on campus with which we were dealing had little or no experience with this type of application. Because the system is a hybrid of custom software, standard software, and several types and makes of equipment often requiring specially designed hardware, deliveries were quite poor, anywhere from days to months late.

Another significant problem is in the area of documentation. The uniqueness of the system, the fact that the vendor was engaged in several custom systems at the same time and that documentation did not have the
highest priority on the vendor’s schedule made it a difficult item to get. We had, however, built in an automatic hold-back of payments tied to documentation. The firm is expanding its sales, and good documentation has become vital. In light of this the vendor’s priorities have changed, and a full-time technical writer has joined the firm. In addition to the vendor’s documentation, there are FLS’s procedures manuals and users’ guides which must be more extensive than would have been required in a standard system.

Another important problem in a hybrid system such as the school’s is maintenance. With such a variety of equipment and special software, trying to establish a set of maintenance arrangements with the different firms could have been a nightmare. However, GEAC is also providing a number of facilities-management operations for users of their systems and has a fairly good maintenance staff. FLS has negotiated a maintenance contract with them. This arrangement will also provide access to changes and enhancements to the basic system as they are developed.

Although there have been frustrations and problems with design and implementation, the results to date seem to indicate that the flexibility obtained and convenience of a fairly powerful in-house system outweigh the problems. Some of these advantages can be seen in the ways that we have begun to use the system.

Presently its most important function is a link between the card reader and the line printer and the university’s IBM 370. This gives on-site access to both the HASP job stream and the High Speed Job Stream for batch jobs. We have dedicated disk space at UTCC. On this we continue to build up data bases for administrative, research, and instructional needs. The simulator for the first-year programming problem is stored in this way, as are the files for first- and second-year course assignments and the FLS library’s data files. The obvious advantages are convenience and saving of time. Until a year ago, a ten-minute walk across the campus was required, often in inclement weather.

In the spring of 1974 we incorporated the minicomputer as a model in the course on file structures. The circulation system of our own library is being redesigned by the class, hypothetically, to operate on-line using available hardware, principally our data collector and our mini. The university library’s Sigma 6/7 will be used for back-up and off-line functions. At the time of writing, this course is still in progress so it is a little premature to evaluate it, but enthusiasm is high among the participants. Within an environment of real constraints, students are gaining experience in systems design, systems development, record and file design, compression coding and hash coding techniques. This approach is making the principles of file organization and information handling gained in the classroom and from the literature much more meaningful.
Not to be overlooked is the usefulness of the mini in providing a local listing capability. This is used by students and staff alike to make listings of programs for debugging and proofreading. Those who are doing calculations of a statistical nature on data stored on punched cards are finding this listing facility invaluable for checking over the data for keypunching errors.

A somewhat novel use of the mini is in operation as a daily notice board, popularly known as the SIN-board (Selective Information Notification). Closed-circuit television screens strategically located in the school's entrance lobby, the FLS Library, and lounge areas flash notices of topical interest. The announcements cycle through at a readable speed, and recycle throughout the day. The texts are stored on the mini, and it is the mini which controls the cycling. Updating may be done as frequently as needed. SIN has implications for library settings where it could serve as a dynamic notice board. In pre-SIN days notices were duplicated and distributed in as many as 300 copies. SIN is living evidence of our fight against paper pollution.

Other applications programs are being developed for the mini. At this point, however, most of the work has been on support software needed for later applications and the expanded communications functions.

The psychological benefits of an in-house system are being realized. The in-house lab helps students to overcome the man-machine interface syndrome so many of them experience at first. With a minimum of direction, students submit their own jobs, tear off their own printouts, and interrogate the central computer for queue status information about their jobs via CRTs. The atmosphere is very informal in our data processing lab, and this encourages a natural curiosity about the relationships between the components, the card reader, the printer, the CRTs, the typewriter terminals, the disk drive and the minicomputer itself. Permanent displays on the walls of the lab to describe the configuration in detail and to explain its relationship to outside systems are planned.

The students have lived through some instances of machine failure. An upsidedown deck in the card reader would jam the printer until software modifications were made to identify and sidestep the mistake. Starting the system up after a software failure is something a number of us, staff and students, can now do with moderate success and considerable satisfaction if and when we succeed. Paper jams on our somewhat temperamental printer are another difficulty we are all learning to deal with. The card reader, too, sometimes balks at swallowing our decks. Students see members of staff coping with these kinds of problems, which contributes to a reduction in the fear of the machine.

The lab is large enough to accommodate a number of work tables and
usually these are spread with printouts. Teamwork is encouraged both officially and by the atmosphere. Informal instruction is fostered too. Often one of the heads in a huddle belongs to a teacher.

It is hoped that the future use of the system will bring applications to a more sophisticated level. For research the flexibility of the system and variety of hardware will be able to satisfy individual needs and will provide an opportunity for comparative studies. For students there will be exposure to a variety of alternatives for library applications.

We have already described one class project using the mini. Many other projects suggest themselves for classes in the years to come: taking a different file organization approach to the same circulation system, working on a different subsystem of the FLS library's operation or developing a new service to promote use of available data bases. Developing actual working systems for our library or prototypes of systems gives students an opportunity to observe and evaluate automated systems in action. The systems available for observation should be similar in equipment and sophistication to systems frequently found in modern libraries. Judging from the current thinking of some of the OULCS member libraries, the general pattern for the future will be network tie-in supplemented by an in-house minicomputer. One library in Ontario is already in the process of purchasing a system like ours. Our graduates will go to their future jobs with some minicomputer familiarity. We are not clairvoyants, but current trends seem to indicate that using the mini as a local processor in a network is the direction many libraries will be going. It is as much good luck as good management that we have a system that anticipates this trend. Having it, we intend to exploit our good fortune.