DOCUMENT RESUME

ED 363 222	HE 026 832
TITLE	Challenges and Opportunities of Information Technology in the 90s. Track VII: Managing Applications and Technology.
INSTITUTION	CAUSE, Boulder, Colo.
PUB DATE	91
NOTE	73p.; In: Challenges and Opportunities of Information Technology in the 90s. Proceedings of the CAUSE National Conference (Miami Beach, FL, November 27-30, 1990); see HE 026 825.
AVAILABLE FROM	CAUSE Exchange Library, 737 Twenty-Ninth Street, Boulder, CO 80303 (individual papers available to CAUSE members at cost of reproduction).
PUB TYPE	Speeches/Conference Papers (150)
EDRS PRICE	MF01/PC03 Plus Postage.
DESCRIPTORS	Academic Libraries; *College Administration; College Planning; Communication Audits; Computer Networks; Computer Software; Higher Education; *Information Management; *Information Networks; *Information Systems; *Information Technology; Microcomputers; Online Systems; School Registration; Student Records
IDENTIFIERS	Boston State College MA; CAUSE National Conference; Image Transmission; University of Iowa

ABSTRACT

Seven papers from the 1990 CAUSE Conference Track VII: Managing Applications and Technology are presented. Authors describe how colleges and universities are incorporating emerging technologies into their campus environments: hardware; software; and procedural techniques. Papers and their authors are as follows: "The Iowa Student Information Services - A Distributive Approach" (Jerald W. Dallam, Marion L. Hansen, Stanley Podhajsky, and Dennis Preslicka); "U-BUY Online Requisitioning - One Giant Step" (Joseph Harrington and David McCormack); "Small-Scale Document Imaging" (Daniel V. Arrington); "Client-Centered Strategic Planning" (Charles Bandy); "A Degree Audit System Implementation in a Distributed Environment" (Bruce L. Rose, James Armstrong, and Vitaly Zavadsky); "Customized Tools: CAAD System (Computer Aided Application Development)" (Wayne Ostendorf and Frank Maly); and "A Common Interface to Multiple Applications" (Candace Wilmot and Catherine Salika). (GLR)

CAUSE

Challenges and Opportunities of Information Technology in the 90s

Proceedings of the 1990 CAUSE National Conference

TRACK VII

MANAGING APPLICATIONS AND TECHNOLOGY

November 27 - 30, 1990
Fontainebleau Hilton Resort and Spa
Miami Beach, Florida

CAUSE90
MIAMI BEACH
NOVEMBER 27-30, 1990

TRACK VII

MANAGING APPLICATIONS AND TECHNOLOGY

Coordinator: Reid Christenberry, University of Georgia

Papers in this track describe how colleges and universities are incorporating emerging technologies into their campus environments: hardware, software, and procedural techniques.

The Iowa Student Information Services - A Distributive Approach

This presentation deals primarily with how The University of Iowa implemented a distributed Student Information System. This is a user-friendly interactive system that enables students to process their own registration request through a computer or terminal in one of 22 Instructional Technology Centers (ITCs) on campus or terminal that is connected to the system, including dial-up.

Jerald W. Dallam
University Registrar
The University of Iowa

Marion L. Hansen
Associate Registrar, Systems and Services
The University of Iowa

Stanley Podhajsky
Associate Director, Administrative Data Processing
The University of Iowa

Dennis Preslicka
Senior Data Systems Manager, Administrative Data Processing
The University of Iowa

The Iowa Student Information Services - A Distributive Approach

Where we were

Prior to 1978 the UI had an arena type registration system. In the Fall of 1978 on-line registration was implemented where trained terminal operators were used to enter the course information.

Transition

The impetus to develop the Distributed Registration (DR) system came from a desire on the part of Univer´´ty planners to provide a simple, flexible system that gave the students more responsibility and made it more convenient for them to register.

Originally University personnel considered developing a touch-tone system, but because of the high cost of implementing such a system and the fact that the University faced budget constraints, the idea of a touch-tone system was delayed.

Following a discussion between the Registrar, Administrative Data Processing (ADP) personnel and the director of the Office of Information Technology (OIT), a decision was made in January 1989 to make use of the University's 22 existing Instructional Technology Center (ITC) facilities in developing a new registration center.

We went with this concept because the terminals were available, the communication network was basically in place, and the monitors were in the ITC's. We felt we could implement the DR system at approximately one-third the cost of a touch-tone system and that the system would pay for itself over the next two years, due to reduced need for staff in the registration center.

This concept also allows us to provide many other services to students in addition to registration, such as address changes, campus employment opportunities, messages to and from Registrar and student. Because this system is much more than a registration system the name chosen for it was Iowa Student Information Services (ISIS).

Many issues had to be discussed and resolved in order to design, code and implement a system of this magnitude. Perhaps the major concern was the preservation of the Universities mandatory advising system. The fact that we had been in the online registration environment for many years allowed us to go on a "fast-track" for the system development. Because many of our files, data bases and software were unchanged or only slightly modified, the transition to ISIS was a relatively smooth one.

The biggest change for ADP was that we normally develop systems for use by trained CRT operators. The ISIS system required that we follow a different approach and develop a "user-friendly", menu driven system with help screens that even a novice operator could use. We worked closely with staff at the WEEG academic computing center, who were more familiar with this type of system.

The timetable of DR was as follows:

January, 26, 1989	Final decision to implement the DR system
March 1, 1989	System design completed by ADP and Registrar
April 3, 1989	Program development begins
October 2, 198	ISIS training for registration personnel
October 25, 1989	Test registration for 100 randomly selected students
November 1, 1989	ISIS system installed in production CICS; test students allowed to register for Spring
November 15, 1989	Start of registration for spring 1990 session: 2400 students allowed to use system

Where we are

At the completion of the early registration in December 1989 it was clear that the system was a success, response to a questionnaire to the pilot group was very positive, full implementation was scheduled for summer and fall 1990 early registration in April 1990. The registration was very successful and while students were not required to use ISIS themselves 72% of them did their own registrations.

With all the other services provided by ISIS, the system is used throughout the year. Of course the peaks are during registration, but we get 20 to 100 address changes daily and the employment opportunities are used daily. Also students can check registration information at any time.

One concern of the DR system is that early registration usually happens to fall during the busiest times of the semester, when many students are using the ITC's to complete papers and research projects. To address this issue, ITC's are staying open longer and scheduled registration times were extended into the evening and weekend hours.

The ISIS system is available around the clock, except for 15 minutes sometime between 1 A.M. and 6 A.M. However most ITC's are only open between 6 A.M. and 12 midnight. During early registration periods students are scheduled to register from 8 A.M. to 9 P.M. on Monday through Saturday. They are currently allowed to register or change their schedule at any time after their registration time until the start of the term. No time limits are placed on the students when they are at the terminal.

During summer orientation sessions all new students register themselves at one of the ITC's. It is hoped that at a date in the near future this will lead to near 100% participation in the use of the ISIS system.

Since the system is used by the registration center personnel, all students are registered through the system. This has led to greater accuracy and consistency throughout, since many enhancements have been added to the new system.

Implementation of the DR system was possible through a combined effort of the Registrar's Office, ADP, WEEG Computing Center and Telecommunications. While not unique to the University of Iowa, the ISIS system is a relatively new concept and the University of Iowa is one of the pioneers in the area.

Future

The future of ISIS is very exciting. With several areas of use either being programmed, planned or discussed, it will constantly be developing into a more encompassing student information system. Perhaps in the near future students will be able to handle all of their administrative dealing through the ISIS system.

In January 1991 ISIS will be used by the registration center personnel to do late registration and drop/adds. This service will be extended to the general student population in the June 1991 session. Other services being discussed are requests for transcripts, degree audits, degree applications, admissions applications and additional financial aid services.

We at the University of Iowa see the future of ISIS as almost unlimited.

Preparation for Registration

The University of Iowa registration system allows students to enter their own registrations via personal computer or computer terminals located at 22 Instructional Technology Centers around campus.

There are four things to keep in mind as we view this registration process. One, The University of Iowa had an effective online registration system in which part-time computer terminal operators entered the registrations for the student at a centralized registration center. Two, mandatory advising is in effect for all colleges except the College of Business Administration. Three, the Iowa Student Information Services (ISIS) system is in effect until the first day of classes. And four, the screens and response messages were designed with the aid of our Academic Computing Office (WEEG) to ensure the use of terminology and screen responses that other students would have encountered in other student-used programs.

Preparation for a registration period begins with the printing of the Schedule of Courses for that particular semester which is approximately six weeks prior to the scheduled registration. At this time, registration forms are printed for all students registered during the current semester. These forms contain demographic data, a course listing area, as well as the student's registration time and unique registration number. These registration forms are mailed to advisers; therefore, the registration number becomes synonymous with adviser approval since the number is issued to the student only by the adviser. A degree audit is sent to each student and adviser also.

At this time, a session segment containing the registration time and number is added to the student database. As registration forms are produced for new and returning students, whether in a batch or online mode, a student/session segment is added to the student database.

So for demonstration purposes, the student has met with adviser, made course selections, and is ready to process registrations at the ITC of choice. The Schedule of Courses contains information on the open hours for ITC's and Registration Center as well as the instructions on how to access the ISIS system through the various computer devices available. Students are encouraged to review these instructions, check eligibility to register, and availability of courses prior to their assigned registration time. By assigning registration times, senior and graduate students have first priority on course selections and also distributes the registration activity over a 14-day period. Times are assigned from 8 a.m. until 9 p.m., Monday through Saturday. Most ITC's are open until midnight and the ISIS system is available 24 hours a day.

ISIS Registration System

There are basically, three levels of student security for entry -- the student social security number, a password, and the registration number -- the latter is required only for the original registration screen entry. If this is the first time a student has been eligible to register, the password is entered by the computer as the student's birthdate. The student then must change it to a password of choice as long as it is one to eight digits in length. This is stored in the system but visible to no one. If a student forgets the password it can only be reset to birthdate by going to the Registration Center. If a student loses the registration form, a new registration form or number will only be issued through the adviser.

As we progress through the registration process, remember that sample screens are printed in the Schedule of Courses and Help screens are provided in the ISIS system for most screens.

If the student enters a social security number that does not exist on the student database an error message is given and any further action is prohibited. If the student enters an incorrect password, five attempts are allowed. On the first two attempts, a gentle reminder is given. On the third and fourth attempts, a more forceful message reminds the student that only five attempts will be allowed. On the fifth attempt, they are exited from the system. We do not block reentry at this time.

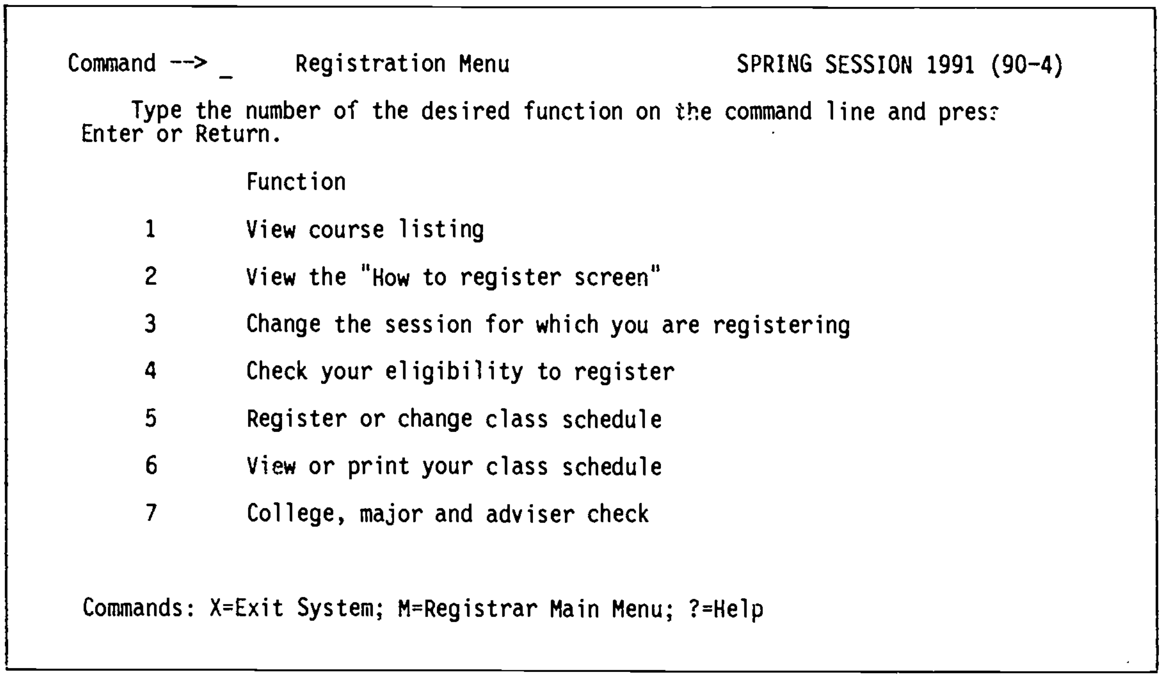
When the student has entered the correct birthdate as password and this is the first entry into the ISIS system, the student is required to change password. For security reasons, the password is nondisplay. For this reason, the password must be verified by reentering. The new password will remain in effect until the student makes a change and the student will need to enter the password each time the system is used. This is the student's security of his/her registration and other data.

The Current Location Screen is next displayed. This information is used for two reasons -- the printing of a copy of the class schedule after registration is completed and the collection of statistics on the activity through the various locations. Because some computer devices are direct cable, some are through communications lines such as Gandolf, and some are via phone, the exact locations cannot be captured by the mainframe computer.

The ISIS Main Menu is now displayed. It is anticipated that other office will add to the functions listed. At the present time, the Financial Aid/Student Employment Menu is the only non-Registrar menu at this time.

The Registrar Main Menu and Registration Menu are requested. This forces the Messages/News Screen if messages exist. These nearly always exist, since Registrar's Offices always have information to dispense. This is used mainly to remind students of address changes, inform when validation sticker will be mailed, and also inform of the session in effect. This comes into play particularly during our April registration when students may register for both the summer and fall semesters.

Most of the functions viewed on this menu may be viewed by the student in the weeks prior to registration time and some of the functions, such as degree applications and transcript requests are not available as yet. Obviously, prior checking of registration eligibility and course availability will help the process.

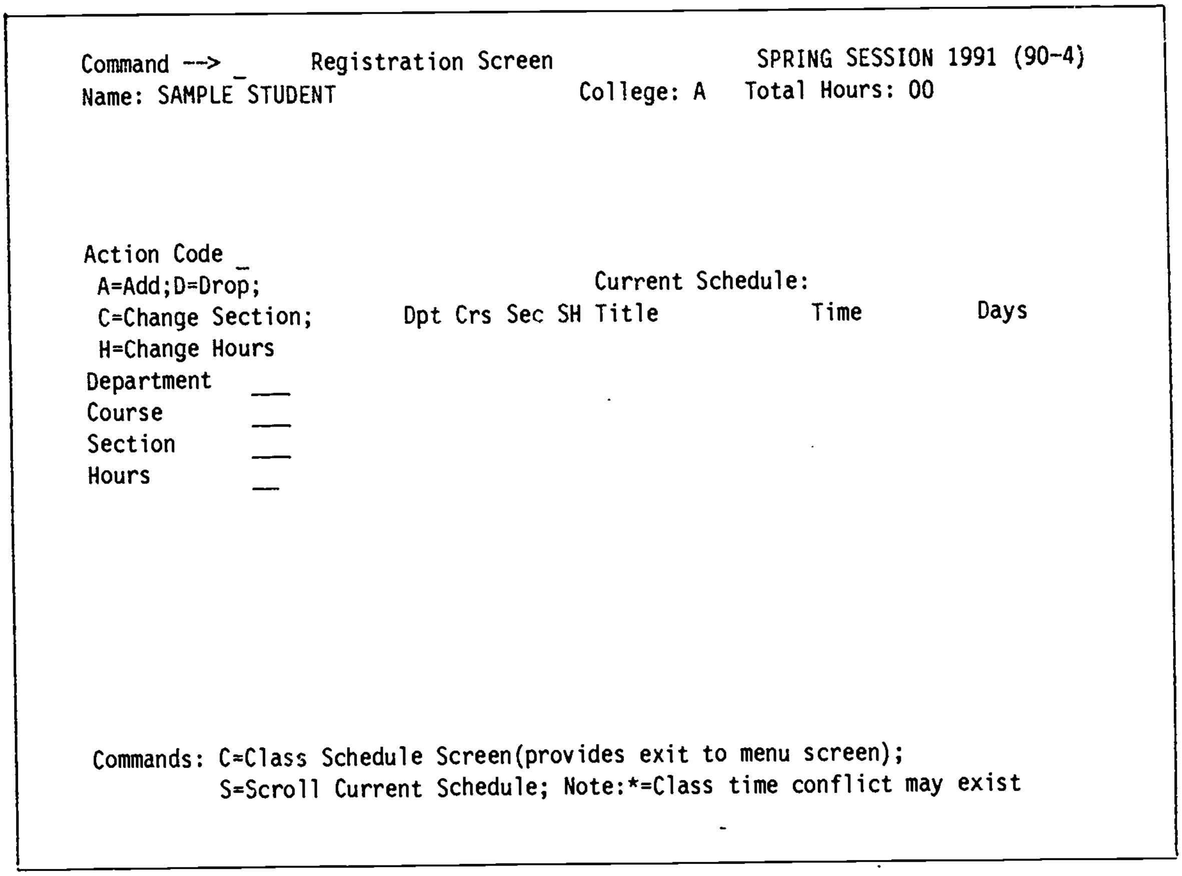


```
Command --> _        Registration Menu                  SPRING SESSION 1991 (90-4)

     Type the number of the desired function on the command line and press
  Enter or Return.

                  Function

          1       View course listing

          2       View the "How to register screen"

          3       Change the session for which you are registering

          4       Check your eligibility to register

          5       Register or change class schedule

          6       View or print your class schedule

          7       College, major and adviser check


  Commands: X=Exit System; M=Registrar Main Menu; ?=Help
```

Since our purpose is registration, the Command Code is 5 is entered and results in the Registration Number Entry Screen. This is a four-digit number and since it is on the registration form which the student received from adviser, three attempts at entry are given. Here again, the first attempt is gentle, but on the second attempt the warning is more severe and contains the statement of consequences.

Successful entry of registration number checks eligibility to register. If a student is ineligible for various reasons, the messages are designed to give specific reasons and specific actions. When a student is attempting to register prior to the correct registration time, the response indicates the correct registration time. When the student has a university bill that is overdue, the response indicates a referral to the Cashier's Office. Also the University has a measles immunization requirement which must be met. Since the ISIS system is reading all the support files directly, a clearance of any of these not-permit actions can be made in the appropriate office and instantaneously be reflected for registration purposes. In the case of indebtedness, the file checks whether the existing indebtedness is actually due or has just been billed. If the not-permit exists, the student is not allowed to proceed further in the registration process. Support files do have the capability to make exception overrides. For instance, an earlier registration time request might be granted for an individual student and therefore a new registration time will be entered on the student database.

Students who are eligible to register are taken directly to the Registration Screen. At first glance, this appears to be a fairly simple screen; however, it will quickly be filled through the registration process. Messages from course action will be displayed at the top of the screen. As courses are accepted for entry on the left of the screen, they will be displayed on the right side of the screen by department and course in ascending order The left corner is reserved for the open sections display.



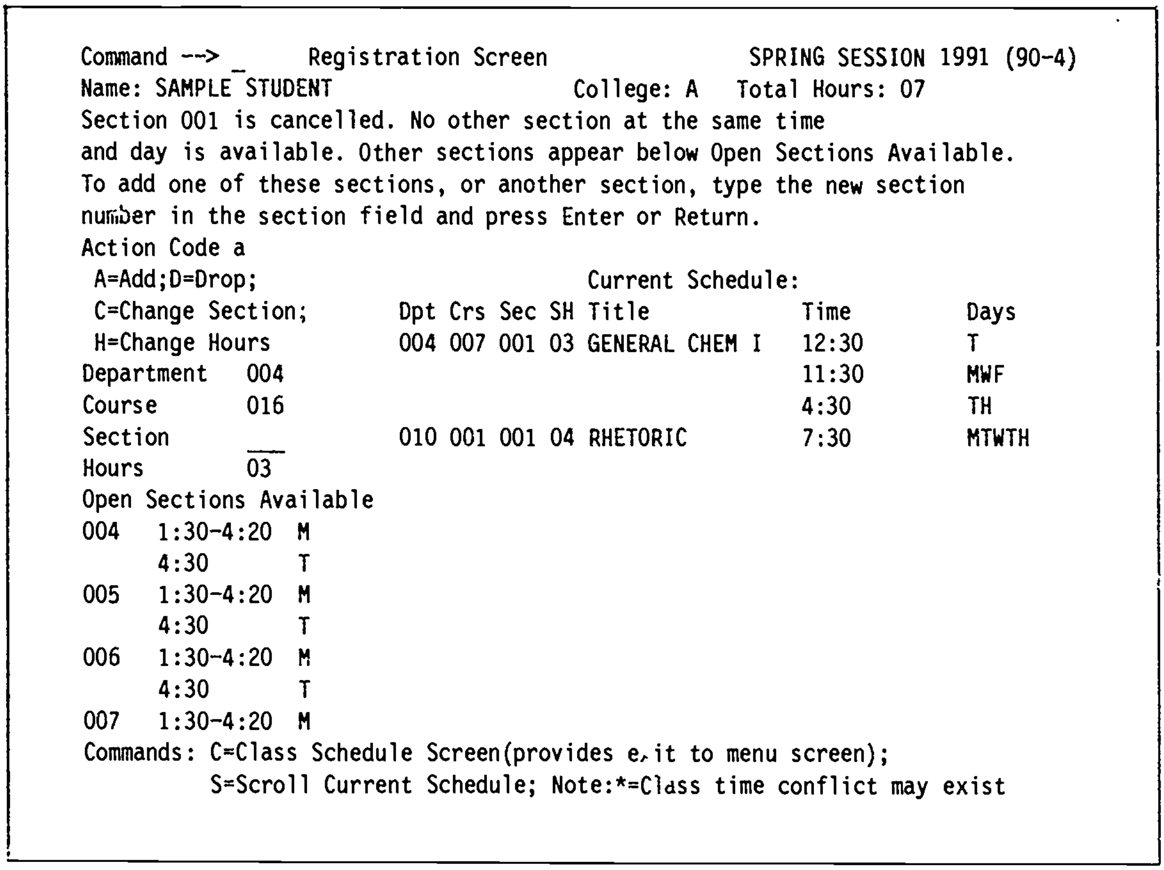
```
Command --> _      Registration Screen              SPRING SESSION 1991 (90-4)
Name: SAMPLE STUDENT                      College: A   Total Hours: 00




Action Code _
 A=Add;D=Drop;                            Current Schedule:
 C=Change Section;      Dpt Crs Sec SH Title             Time         Days
 H=Change Hours
Department    ___
Course        ___
Section       ___
Hours         __






Commands: C=Class Schedule Screen(provides exit to menu screen);
          S=Scroll Current Schedule; Note:*=Class time conflict may exist
```

Registration

The action code must be entered for each entry and thus set in place the various validity checks. The University of Iowa has historically used the department, course, and section number for registration in lieu of an index or sequence number. To begin course entries, an A for Add action is entered and the three-digit department, course and section numbers and the two-digit hours. If this is a valid action and therefore the course is accepted and displayed on the right. Each action is confirmed or an explanation as to why the action is invalid.

If the student wishes to change the section, the action code of C is entered and the department, course, section, and hours to be changed. Upon entry, the department, course and hours remain and the section is blank with the cursor positioned for section entry. The message above reminds student of current section and tells student to enter new section, the new section is entered, the confirmed message is displayed and the new section is listed at the right side of screen. This is double bright on most computer devices.

One of the very helpful innovations in the registration system is our display of Open Sections Available. When a course is closed due to maximum enrollment, the closed message is given at the top of the screen and reminds the student of the original section requested. Here again, the department, course, and hours remain and the cursor is positioned in the section field. The lower left displays the first open sections of the course to a maximum of seven. If there is a section open at the same time as the original section requested, this section is displayed only. If additional sections need to be viewed, the student can go to the Course Listing Screen for further selection.



```
Command --> _      Registration Screen              SPRING SESSION 1991 (90-4)
Name: SAMPLE STUDENT                     College: A   Total Hours: 07
Section 001 is cancelled. No other section at the same time
and day is available. Other sections appear below Open Sections Available.
To add one of these sections, or another section, type the new section
number in the section field and press Enter or Return.
Action Code a
 A=Add;D=Drop;                           Current Schedule:
 C=Change Section;      Dpt Crs Sec SH Title            Time         Days
 H=Change Hours         004 007 001 03 GENERAL CHEM I   12:30        T
Department   004                                        11:30        MWF
Course       016                                        4:30         TH
Section      ___        010 001 001 04 RHETORIC         7:30         MTWTH
Hours        03
Open Sections Available
004   1:30-4:20  M
      4:30       T
005   1:30-4:20  M
      4:30       T
006   1:30-4:20  M
      4:30       T
007   1:30-4:20  M
Commands: C=Class Schedule Screen(provides e,it to menu screen);
          S=Scroll Current Schedule; Note:*=Class time conflict may exist
```

Special Permission Courses

Courses that are designated as requiring special permission for registration are indicated in the Schedule of Courses and entered with a series of codes that check the type of student requiring special permission. For instance, business courses are coded to allow students with business majors to register but students other than business majors need to acquire special permission. Departments may select three ways in which to give special permission. One, and obviously preferable, is to enter the special permission course on the student database. At the present time, all dean's offices and some departmental offices have this capability. Second, a series of random numbers may be issued for dispersal by the instructor or department. These random numbers are keyed to the course and when a student uses a special permission number, it is flagged as used on the course database to prevent its reuse. Third, an instructor's signature can be entered on the registration form and processed through the registration center.

If a student has entered a registration in a special permission course for non-business students and the course is not entered on the student database, the special permission number area is displayed for entry. The message is specific as to what office needs to grant this special permission. If the student has obtained it, the special permission number can be entered and registration in the course is accomplished. If the student database had contained the special permission course, the student would have been accepted into the course immediately upon entry. If this were a course which requires an instructors signature, the course would not be accepted through ISIS.

Independent Study courses have as a section number the instructor number as assigned by the department. This is used for producing separate class lists for these courses and also for producing instructional data information. The instructor numbers are added to the course database prior to registration. When a student registers and enters the instructor number, the validity of the number is checked, but the instructor number is not flagged because it needs to be reused by the instructor for additional independent study students. The registration system requires a valid instructor number as a section number for this type course.

Hours Change

If the student wishes to change hours, an action code of H is entered and the department, course, section and hours to be changed. Upon entry, the department, course and section remain and the hours are blank with the cursor positioned for hours entry. The message above reminds students of current hours and tells student to enter appropriate hours. If new hours are accepted, the confirmed message is displayed and the new hours listed at the right side of the screen. However, University regulations allow only graduate students to take a course for reduced hours. Other colleges allow students to audit or register for zero hours with special permission. Even when an accepted audit hours is made, the student is informed that the fee hours assessed for the course is the minimum for which the course is offered. Special permission for audit is processed in the same way as other special permission courses.

Problem Courses

Certain courses have components that need to be linked together for registration. An example of this would be one of our math courses which has an option of lecture and an option of discussion. Since separate class lists and reports on instructional effort need to be produced, the students are required to register in both components of the course. By using a set of codes (affectionately called Problem Course codes), the components are linked together so that when a student registers in the first component, a message is given on the selection of the second component. We display the choices of the second component in the lower left corner of the screen. Similar to a section change, the department and course remain so the only entry required by the student is the new section and hours. Registration in the course does not occur until both components are selected; however, a place has been reserved in the first component. When entry has been accepted, you will note that both entries appear on the right side of screen. Conversely, when we drop one component, both components are dropped.

Drop Courses

The system does allow the students to drop all their courses. This constitutes a void of total registration and the message confirms this fact. Since we view the electronic record as a contract for registration and fees, the voiding of this contract is important to confirm.

Registration Confirmation

All screens have had the capability of exiting the system so a student can stop and start at any point during the process. One exception to this is the registration screen. Because we wanted to officially confirm the registration and give the student an opportunity to request a printed copy, the registration screen must be exited through the Class Schedule Screen. This screen also gives one more notice of any course conflicts. The Class Schedule Screen is set up so that the printed copy is a screen dump. If the ITC in which the student registered does not have a printer, the student can go to any other ITC or to the Registration Center and use the Command 6 (View and Print Your Class Schedule) on the Registration Menu.

Additional Registration Features

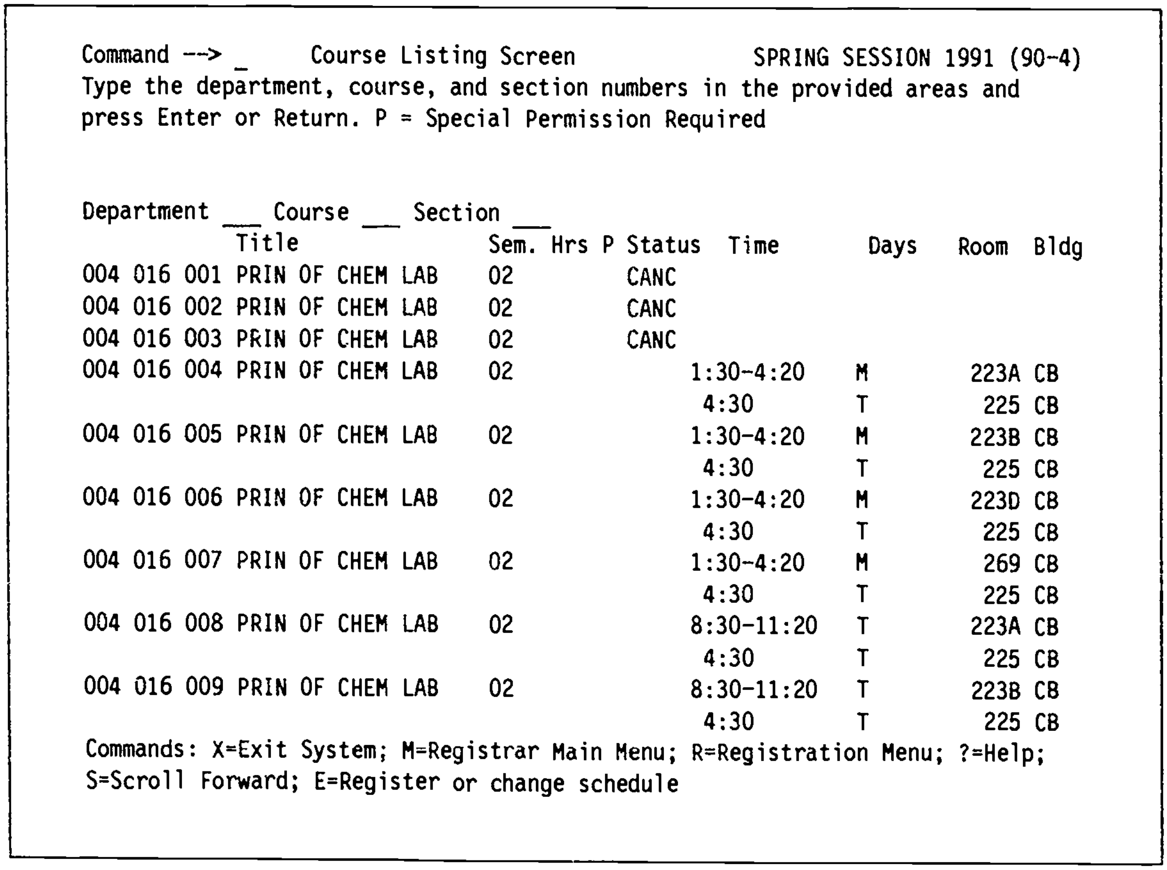
Once students have completed registration, they may enter the registration system at any time to make updates or check on course availabilities. The security for entry would be the social security number and the password entered by the student. This password may be changed at any time as well. Registration changes can be made until 5 p.m. the night before classes begin for that session; however, viewing can be done at anytime.

As registrations occur in which a class time conflict exists, an asterisk appears to the right of the entries that are in conflict. Because the University of Iowa has valid time conflicts between certain courses, it was decided to inform the student and not to prevent the conflict to exist. This is an academic matter, but since we have programmed to identify the conflict it would be very easy to prevent the conflict. It was also decided not to prevent multiple registration in the same course. This again would be easy to prevent, but is based on academic considerations. The student obviously has the option of correcting the conflict by adding or dropping courses involved.

Undergraduate students who wish to register in a graduate level course must obtain special permission from the instructor of the course. A message to this effect appears at the top of the screen and the special permission number is requested. Similar to the special permission courses, if the course appears on the student database as permission granted the message would not occur in the registration process.

Maximum hours of registration are in effect for all colleges. When a student exceeds this maximum, a message to this effect appears at the top of the screen. Exceptions to maximum hours are granted only by the deans' offices and since these offices have the student database, an override can be entered. Here again, if the exception exists, the message would not appear.

The Student Comment Screen was designed as a suggestion box for students to leave comments on the registration procedure. Our original fear that this might be a "gripe" screen was not confirmed. Student comments helped us to identify good and bad points of our procedures. We could also identify any ITC scheduling problems as well as assess the helpfulness of monitors and coordinators in these centers. Approximately 95 percent of the comments were favorable and mentioned the fact that registration lines were eliminated, the hours of registrations were more conducive to student schedules, and made preplanning and course selections much easier. The Student Comments are delivered to the Registrar's Office on a nightly basis and these are reviewed by personnel in the Registration Center and in the WEEG Computer Center. This was an important part of our registration process.



```
Command --> _      Course Listing Screen           SPRING SESSION 1991 (90-4)
Type the department, course, and section numbers in the provided areas and
press Enter or Return. P = Special Permission Required


Department ___ Course ___ Section ___
            Title               Sem. Hrs P Status  Time       Days   Room  Bldg
004 016 001 PRIN OF CHEM LAB    02         CANC
004 016 002 PRIN OF CHEM LAB    02         CANC
004 016 003 PRIN OF CHEM LAB    02         CANC
004 016 004 PRIN OF CHEM LAB    02              1:30-4:20   M      223A CB
                                                 4:30       T       225 CB
004 016 005 PRIN OF CHEM LAB    02              1:30-4:20   M      223B CB
                                                 4:30       T       225 CB
004 016 006 PRIN OF CHEM LAB    02              1:30-4:20   M      223D CB
                                                 4:30       T       225 CB
004 016 007 PRIN OF CHEM LAB    02              1:30-4:20   M       269 CB
                                                 4:30       T       225 CB
004 016 008 PRIN OF CHEM LAB    02              8:30-11:20  T      223A CB
                                                 4:30       T       225 CB
004 016 009 PRIN OF CHEM LAB    02              8:30-11:20  T      223B CB
                                                 4:30       T       225 CB
Commands: X=Exit System; M=Registrar Main Menu; R=Registration Menu; ?=Help;
S=Scroll Forward; E=Register or change schedule
```

We now want to look at the additional screens that are accessible through the Registrar Menu. The Course Listing Screen is available to students, deans' offices and departmental offices and gives the up-to-date status of courses. Note that it can be access to begin with a department, with a particular department and course, or with a particular department, course, and section. The last option is especially helpful when the student wishes to see additional options to the Open Sections Available section of registration screen. In addition to the usual necessary items, such as title, semester hours, meeting time and place, the special permission code and status of course are indicated. A complete definition of these codes are included on the Course Listing Help Screen. Remember this is viewable prior to registration.

Some courses are indicated as Departmental Wait Listing and others as Wait Listing. Students are referred to the department when the department maintains its own wait lists. The Registration Center maintains wait lists on those courses in which the department has requested our monitoring. Special screens are provided in the Registration Center for this purpose. Students are not allowed to be placed on a wait list unless the total course is closed and a maximum of two courses per student can be wait listed. The time of request is recorded and this information supplied to departments for action.

Support Screens

The College, Ma‿or and Adviser Screen is provided for students to verify their current status. Occasionally students have forgotten their adviser's name or address or a change may have occurred within the department. This screen is useful in making an adviser appointment and obtaining the registration form.

The Change of Address Screen provides a convenient way for the student to meet the University requirement that a change of address be reported within three days. The student can enter a new residing address and an effective date. This change will then go into effect on that date and will be made on all University files, including payroll. Home and Parent Address are provided for information only and the student must submit a Change of Address form to change these, since the volume and procedure is different. This screen is accessible at all times.

Also accessible from the ISIS Main Menu is the Financial Aid/Student Employment listing. This provides employment information on a timely basis and has eliminated student traffic in our Student Financial Aid office. We look forward to other enhancements.

One important support screen is the Special Permission/Enrollment Update Screen which is accessible in departmental office with computer support. From the Registrar Main Menu, the Department Menu can be selected and then the support screen. The department requests the department, course, and section needed to display the status and enrollment figures. The current enrollment is the actual number of student enrolled at the present time, the optimum number is an adjustable figure to control numbers in sections especially during Freshman Orientation, the maximum number is the number to be allowed into the section, and the capacity is that of the room assigned. Optimum and maximum can be changed, but not to exceed the room capacity. In this way, departments can directly control section enrollments.

The same screen is used to record students who have received special permission to register in the course (if required). As the social security number is entered, the student database is checked and responds with the name of the student for the department to verify. This information will now be accessed through the registration process.

By use of a set of security codes, a department is authorized to make changes to only their department courses and their special permission codes. Exceptions can be made by the authorizing department and these are entered through the Registrar Security Control screen.

Summary

The real benefits of this type of registration system is that throughout the process the control is placed in the hands of the person needing to make the decision. The student controls course selections, support offices control the clearing of registration holds, and departmental and deans' offices control the enrollment figures which are under their departmental budgets. Much cooperation is required in the Schedule of Courses, but once this is accomplished, no phone calls or paperwork impedes the process of registration. If a department wishes to work directly with the registration of the students the system will allow this as well.

We finally have a registration process which occurs while we sleep!!

MLH/skm

U-BUY - ONLINE REQUISITIONING - ONE GIANT STEP

JOSEPH HARRINGTON AND DAVID MCCORMACK

BOSTON COLLEGE

CHESTNUT HILL, MASS 02167

CAUSE CONFERENCE 1990

CHALLENGES AND OPPORTUNITIES of Information Technology in the 90's

UBUY On-Line Requisitions - One Giant Step!

Abstract:

In the Summer of 1989, Boston College implemented the on-line U-BUY Requisition System. Today, after a year of operation, we have over 600 users in 130 campus departments. The only way to produce a check requisition or a purchase order is thr,ugh the UBUY System. There are no manual routes to bypass the System!

UBUY was our first information system that required on-line preparation of financial transactions in 130 departments. The system incorporated the concepts of a Universal Position System that classified position privileges (including computer security access) and an electronic signature component that controlled system/account access and limits of spending. It represents a significant step as we cross the threshold of campus-wide information systems into the 90's.

We look back and view this development as a 'watershed' application that challenged our users and presented the opportunity for elevating the level of computer literacy and comfort throughout the campus. U-BUY also represents prototype application on which increased campus wide functionality will be provided in the early 90's.

Cause Conference 1990

U-BUY - On Line Requisitioning - One Giant Step!

INTRODUCTION

Over the past year and a half, Boston College has implemented the University Requisition System called U-BUY. This paper/presentation describes the general concepts and system features of U-BUY. The systems development life cycle is outlined over the two years of development along with the problems that emerged.

Finally, there will be a summary of a survey of the University Community to measure the community's reactions to the implementation of the project, the functionality of the system, and the experiences and effects of the first campus-wide system involving the input of critical financial transactions.

AT IS UBUY?

A. Overall System Structure

U-BUY is a menu-driven on-line requisition system with functional capability to do a number of financial transactions.

The UBUY system works in conjunction with other components in our administrative Systems environment. The components and their functions are listed below:

1. Universal Position System

   This system contains a record for every position in the University and it contains the position's campus address, campus phone number, and computer access privileges. It is a part of the overall Human Resources System and it dynamically interacts with the Payroll Personnel System components.

2. University Signature System

   This system defines, account approval, authority for university positions. One type of approval record is a requisition approval. Another type of approval record is a budget transfer approval record. Future types of approval records will be payroll time sheet approval records and personnel action-form approval records.

3. University Accounting System

   As U-BUY requisitions are entered into the on-line system, there is an access of the accounting files to verify that there is a valid account for the transaction and that there are sufficient funds available to make the purchase or payment. If funds are available, then an encumbrance is immediately established and the balance available on the line is reduced.

Critical to the control considerations of U-BUY is the interaction with the University's Financial Accounting System in the functions of:

Requisitioning
Invoice Matching
Application of Credits
Release and Payment

4. University ID System

This system identifies all individuals associated with the university (student, employees, and other defined groups). It keeps track of each individual's current status and grants specific privileges based upon that status. These privileges include (but are not limited to) recreation complex entry, library privileges and administrative systems access. The level of access given within the administrative system is determined as a composite of access given to each of the individuals currently active positions (this includes U-BUY signator access).

B. Overall System Operation

1. Requisitioning

The terminal transactions of U-BUY are accessed through CICS in an MVS environment. The User inserts a PIN number and a Username to sign on to the system. With this information the system knows the Universal Position Number(s) of the signed on User and thereby sees what computer access privileges are available to the position number(s) of the signed-on User. If the employee has access to U-BUY, then a menu is presented. The User selects a function and one or more functional sub-choices are presented.

When a line of information is entered, the system edits the data, verifies the User has access to use this account (or account sub-code line) and has funds available to spend from this account.
(If a User is purchasing supplies from a sub-code line, but has no funds in the sub code line, then the User may exit the process, go to the budget transfer function, and transfer funds from some other line into supplies. Then the User can return to the requisition process and process a requisition on the supplies sub code line).

The User can insert a number of lines, and each line is edited appropriately. When the User is finished entering lines, the system goes to a closing screen which is preformatted accordingly, depending on whether the transaction is a purchase order or a check requisition. The User can change the defaults presented on the closing screen, and can optionally insert comments before marking the transaction as complete by inserting a private password in a screen-darkened area.

2. Approvals

If a line item on the requisition exceeds a unit cost of $500, the requisition is electronically routed to a second appr^ver of this accounr. If there are no lines with a unit cost of ⸗ɔ00 or more then a purchase order is immediately routed electronically to the Buyer in the Purchasing Department. Check requisitions are routed electronically to the Accounts Payable Department of the Controller's Office. Receipts for check requisitions transactions are inserted in a specially designed blue envelope. Campus mail services gives these envelopes express sorting and transmission to the Accounts Payable Department. Some Purchase Orders require approval by buyers before they are printed on a laser printer in the Purchasing Department. Other ṛurchase orders are immediately printed in the Purchasing Department after a User has inserted a password on a requisiticn.

3. Invoice Matching and Payment Release

Accounts Payable reviews receipts and invoices and releases check requisitions which are paid on a batch check run that runs three nights a week.
The Purchasing Department matches invoices to PO's on lines, resolve problems and closes out purchase orders. Invoices are transmitted to Accounts Payable and payments are released against purchase orders.

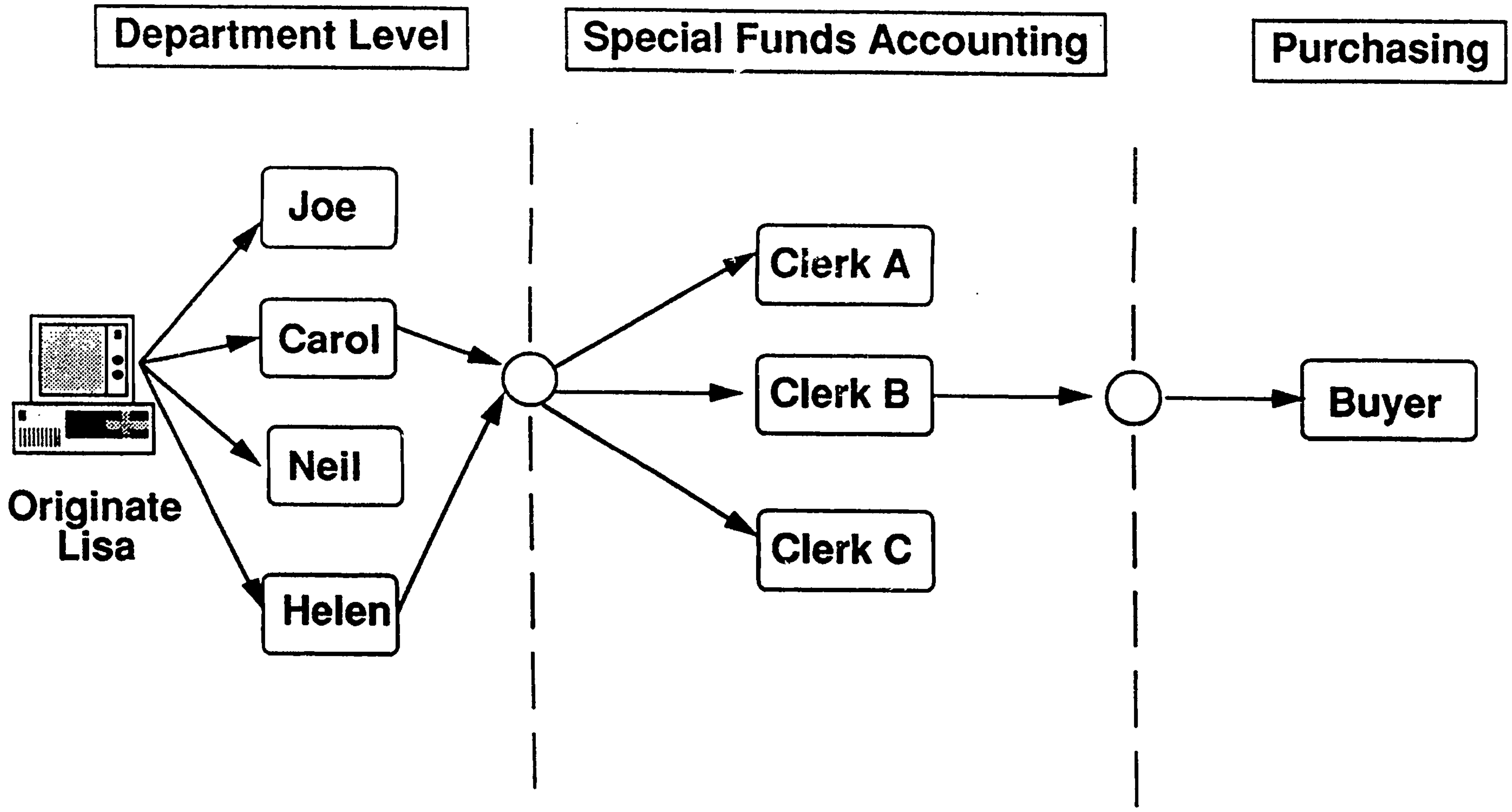
4. Viewing Requisitions

All Users can view their requisitions in a number of different ways.

Unpaid Originator's Requisitions
Paid Originators Requisitions
Unpaid Department Requisitions
Paid Department Requisitions
All Requisitions by an Account Number
Any individual Requisition
Last Year Requisitions

Signator Approval records for requisitions contain the type indicator, the position number, the account number and limits to spending on the specified account. An employee with fiscal responsibility for a number of accounts will have a minimum of one record for each account.

Approval Forwarding



Department Level

Special Funds Accounting

Purchasing

Originate
Lisa

Joe

Carol

Neil

Helen

Clerk A

Clerk B

Clerk C

Buyer

C. Campus Wide Functionality

Authorized Users in any of our 140 departments may originate many different transactions provided they have security clearance for their position, spending authority over the referenced account number, and most importantly, available budgeted funds in the account they are using. The available transactions are:

1. Originate check requisitions.
2. Originate Purchase Orders, Blankets, and Agreements.
3. Originate Requisitions for Internal Services
   - Computer Store Purchases
   - Purchasing Stock Room Requisitions
   - Bookstore Purchases
   - Bureau of Conferences
4. Approve requisitions when more than an originator's approval is necessary.
5. Adjust Blanket Purchase Orders
6. Cancel Requisitions/Purchase Orders under controlled circumstances.
7. Originate budget transfer of funds between 2 six digit accounts or between 2 sub-code lines of an account.
8. View transactions in numerous formats.
9. Receive goods on a purchase order or refer received goods problems to the Purchasing Department for problem resolution.

D. Features of Campus Wide Transactions

- The campus-wide requisitions listed above must be processed in an on-line mode.
- There is no alternative such as the submission of a paper check requisition to the Controller's Office or a paper purchase order to the Purchasing Department.
- There is no alternative such as having the centralized purchasing department enter purchase orders for departments because the originator's position must be established as having spending authority over the accounts that are used.
- Completion of the transaction at the department level performs an instantaneous encumbrance and reduction of balance in the accounts that are used. Real-time means real control!
- When a department originates a purchase order to a contracted vendor and each line of the purchase order is under $500, the purchase order prints immediately on a laser printer in the Purchasing department.
- Purchase orders are processed in conjunction with a commodity table. The commodity table carries the position number of commodity-buyer so P.O. transactions are electronically routed to the proper buyer for review and approval.
- Transactions originated at the departments are electronically referred to the next appropriate approval station.

E. Specific Functionality for Business Offices

The Purchasing Department, Internal Services Department, and the Controller's Office have specific transactions for approving, suspending, and modifying requisitions. Additionally, they have other transactions for matching invoices to a purchase order, processing credits, releasing transactions for payment, requesting additional copies of purchase orders etc.

Buyers in Purchasing have capability for adding extensive text to any purchase order to stipulate contractual terms and conditions.

The Controller's Office

The Restricted Funds Section of the Controller's Office has the functionality to approve or suspend any requisitions drawn on restricted funds such as contracts and grants or agency accounts.

The Accounts Payable Section of the Controller's Office has capability to:

1. Release a payment on a check requisition.
2. Release payment for an invoice on a PO.
3. Suspend check requisitions back to originating department.
4. Modify addressing for check disbursement.
5. Apply credits to specific vendors or requisitions.
6. Maintain addresses for a vendor.

WHY U-BUY?

In the summer of 1986, our survey of the User-community indicated a universal growing displeasure with the paperwork processes involving check requisitions and purchase orders. Everyone was asking for help in controlling the burgeoning paperwork and in eliminating the tremendous time delays in receiving a check for an order or for a reimbursement. Purchase Order Processing took days just to get a purchase order to a vendor. The ordinary check requisition took 14 business days or more. At peak processing periods, it could be 20 days.

Our payments to vendors were late, early payment discounts were virtually non-existent. An order by an administrator for a dozen ball point pens required 2 signature approvals.

Executive management was insisting that functional managers control their budgets - but receiving a budget report 12 days after the close of the month was not conducive to good control. About one fourth for our departments had access to terminal displays of their budgets. Most of our departments, however, were creating "bootleg accounting systems", keeping track of their spending with annotated listing, index cards, or PC spreadsheet programs. The challenge of the 80's was to keep track of their fiscal operations with a "kludgy" set of unintegrated tools.

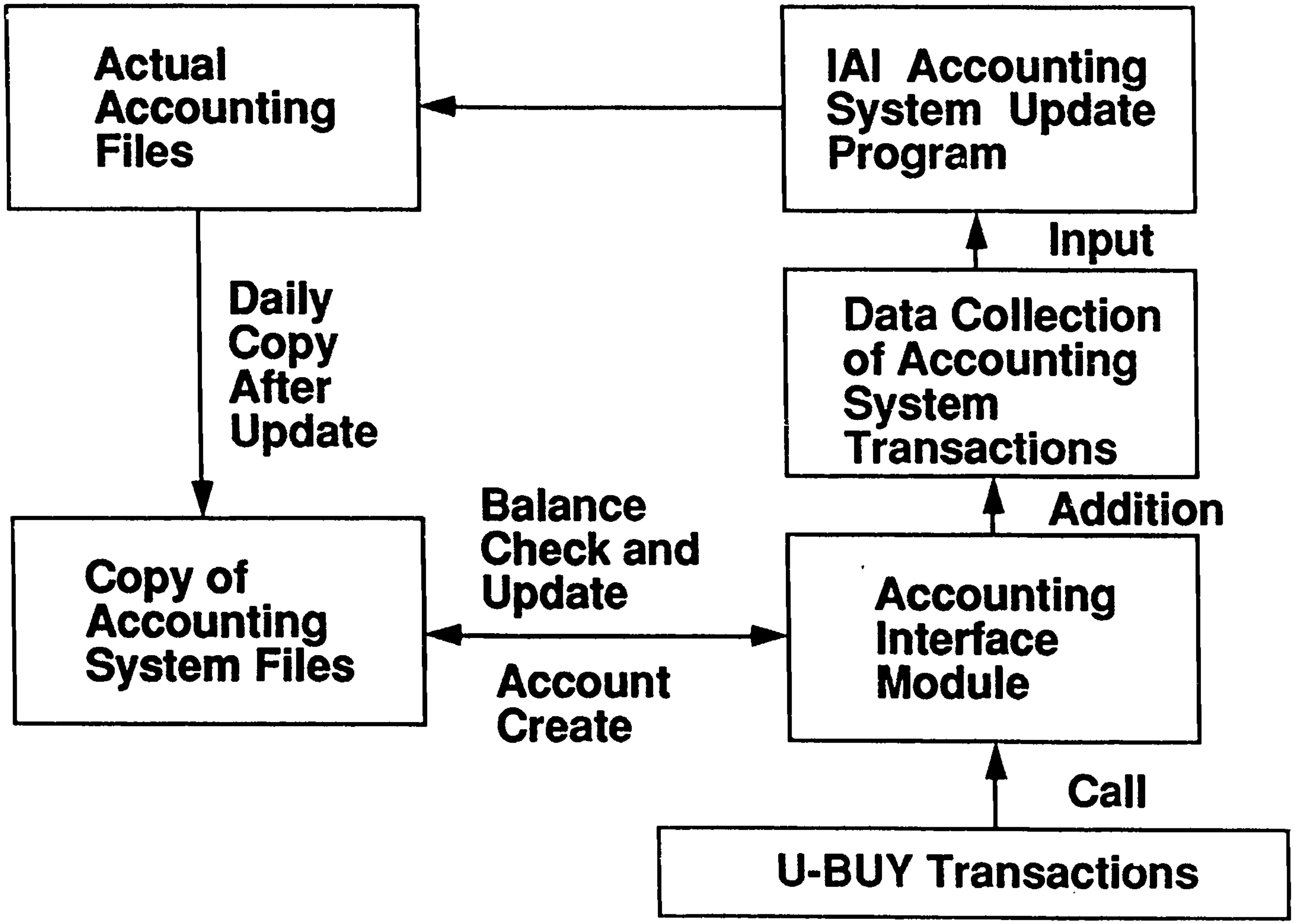
HOW U-BUY?

An annotated chronology will show some of the specifics of how this application was developed and the time-frame in which the product was implemented.

TIME

Fall	1986	Directors of Offices within the Financial Vice President's area met repeatedly with MIS. Result: General Systems Requirements
January	1987	Development of RFI/RFP
March	1987	Review Vendor Proposals
April	1987	Decision to develop product in-house and integrate with our environment.
June	1989	Developed Universal Position System as necessary components to provide a secure system environment. Develop prototype of requisition menus, transactions, and commodity tables. Complete front end design and development.
May	1987	Developed Electronic Signature File
June	1988	Establish Implementation Committees, Security Awareness, Education & Training, Publicity & Promotion, User Documentation and Connectivity. June 1988 All Committees Working
	1989	Financial Systems (Controller's Office), defining positions with account access. MIS - Detailed design and development of back-end components in system. i.e. Invoice Matching Receiving Problem Orders Processing Credits Centralized Laser Printing of PO's AP Release of Payments
Feb	1989	MIS, User Management, Internal Audit, External Audit convened to get total agreement on the matter of electronic signatures.
April	1989	Phase I went live - Next Fiscal Year Purchase Orders; On-line Budget Transfers of the Current fiscal year.
June	1989	Phase II implementation All Purchase Orders and Check Requisitions, and back-end components of system (No internal Charges)
October	1989	Purchasing Stockroom Internal Charges
January	1989	On-line Journal Entries for Charge-back
March	1990	On-line Computer Store Requisitions & Integration with the Info-Technology Inventory/Order system.
June	1990	Completion of U-BUY Year-end processing and special accrual logic
Sept	1990	Bookstore Internal Charges
Nov	1990	Bureau of Conferences Internal Charges

Real-Time Balances
With Account System Integrity



Actual Accounting Files

IAI Accounting System Update Program

Input

Data Collection of Accounting System Transactions

Daily Copy After Update

Addition

Copy of Accounting System Files

Balance Check and Update

Account Create

Accounting Interface Module

Call

U-BUY Transactions

What Were the Problems?

Anticipated:

1. U-BUY would be a whole new way of doing business and would require a great deal of change for a many of our departments and personnel.

2. It would be difficult to coalesce agreement in the Business Offices (Controller, Budget, Purchasing, Interal Audit), on the matter of approvals and electronic signatures.

3. Given the fact that we had no central receiving department, User department receiving would have to be simple with an electronic referral of unusual received purchase orders to a problem resolution function in the purchasing Department.

4. Problem-resolution and invoice matching of purchase orders in Purchasing would be very difficult to program because of the endless possibilities and accounting budget effects.

5. Job re-definition would be necessary in the Accounting and the Accounts Payable departments of the Controller's Office and in the Purchasing Department.

6. Academic department participation in training would be ignored by many.

7. It would be a major task to train 600 Users of the system.

Unanticipated Problems

1. Invoice matching and problem resolution in the Purchasing Department proved much more-time consuming than anticipated.

2. Requirements by the Purchasing Department, Buildings & Grounds representative were much more than we realized in respect to annotating orders and issuing change orders.

3. As an audit trail for information, the system was stamping the transactions with the position number of the employee who did the transaction. We had to change this stamp to include employee ID number for historical purposes.

4. Certain senior-level Users in the Business Offices had exposure to the installation of vendor packages at other universities. The experiences they encountered at other places was translated to negativity in anticipation of what we were about to do. So there were "Black clouds", (these critics have opted for silence or praise since U-BUY was implemented).

5. One function in the Business Office segment ignored our admonitions that the system would dramatically affect their operations and that some restructuring should take place. It's clear to the whole community that this area is a bottleneck that causes unnecessary delays.

Survey of User Community

In July of 1990, we gathered statistics for the previous years' processing under U-BUY. We also surveyed the prime UBUY contact in 25% of our departments (excluding the business offices) and we learned a great deal.

1. We announced U-BUY to the general community in January 1989 seminars. On our survey we learned that nearly a quarter of our respondents questioned their ability to perform sufficiently under a new system when they heard that it would be coming. There was a good deal of anxiety in the User community.

2. It was the Users' perception that under the old paper system it took 14.4 business days to receive a check after the submission of a paper check requisition. With U-BUY, they perceive that they get a check in 7.2 business days after entering an on-line check requisition.

3. With a number of stops along the way for a paper check requisition, and a long delay in receiving a check, phone calls became the usual tool of expediting. In our survey, people reported they saved an average of 2 telephone hours per week, tracking down requisitions, invoices, purchase orders. Conversations between Departments, Purchasing, Budgeting, Accounts Payable resulted in an enormous amount of telephone traffic. If one department saves 2 hours a week, that's a 104 hours per year. When you multiply that by 130 departments it amounts to 13520 hours per a year. But phone conversations are between two people so we can double the number and see 27,040 hours of savings in phone conversations on our campus in one year. That's the equivalent of more than 3800 seven hour days. That is the opportunity for 15 years of increased productivity.

4. 65% of the survey respondents felt their computer and computer system knowledge had risen since U-BUY went live.

5. The survey question: Yes or No in a return to the old manual paper drive system received a resounding 'NO' with one threat of "I'd quit".

General Response by the User Community

The survey coupled with observations and conversations with the User community lead us to believe that the UBUY system has been fairly well received as a friendly system that helps people in the management of their budgets and the operation of their offices. When our plans for the UBUY implementation were announced in early 1989, one fourth of the Users had real reservations about whether or not they could meet the challenge of an on-line requisitioning system. It's a credit to many of the staff of the University that they met the challenge and exceeded their own expectations in a computer environment hitherto unknown.

UBUY spurred connectivity to the main frame and in many ways drove the resource components of Information Processing Support for training and Network Services for on-line connections. More importantly, it stirred imaginations and appetites of our employees who are now asking for on-line Payroll Time-Sheets, Personnel Action Forms (PAF's) and Building and Grounds Work Orders. It's apparent that the UBUY system is accompanied by a heightened awareness and use of information systems as a more viable communications medium.

UBUY as a prototyre, represented a tall mountain to be conquered. However, having done this, the introduction of new on-line functionality for other applications will only be rolling hills and much easier to scale for technology and the User community.

It's necessary to say a few words about attitude in our experience of changing the way we do business. The project implementation was a major step for everyone. There were system bugs and problems, on occasion, but the overwhelming majority of the User community was understanding and patient. Not all, but most were sensitive to the complexity and ambitiousness of what we were all trying to achieve together. The spirit and cooperation of the community was an integral part of this successful UBUY Project.

UBUY is a unique product designed for Boston College. Behind UBUY are various system components and standards that have been carefully designed and developed over the past decade. These pre-existing home-grown components contributed greatly to the success of UBUY. UBUY in a sense was just the frosting on the cake, the sweetest part for all the community to use.

SMALL-SCALE DOCUMENT IMAGING

DANIEL V. ARRINGTON, COAP

UNIVERSITY OF FLORIDA
GAINESVILLE, FLORIDA

ABSTRACT

Document imaging is the process of transforming printed text, pictures and figures into computer-accessible form. While this technology certainly offers dramatic opportunities for enhancing office automation, vendors have been very quick to promote imaging as the ultimate solution for document management problems involving both space and personnel.

A hands-on evaluation of PC-based document imaging was conducted to gain a better understanding of the issues involved. Although we were not successful in our attempt to fully duplicate commercial system functionality, observations from this assessment may be helpful in deciding how imaging should be applied within your organization.

PREPARED FOR CAUSE '90
MIAMI BEACH, FLORIDA
NOVEMBER 1990

Small-Scale Document Imaging

Daniel V. Arrington, COAP

Introduction

Estimates suggest 95%[1] of the 1.3 trillion documents stored in United States offices today[2] are saved in the form of paper and of these, only about 1% are available in various computer formats[3]. At the same time, concerns about declining fiscal conditions are beginning to impact hiring and spending patterns in many higher education institutions. These observations translate into an indisputable need to deal with increasing volumes of paperwork before the situation becomes completely uncontrollable.

Recent advances in automation technology have spurred growing interest in the idea of using computers to resolve increasingly critical problems associated with processing, saving and using paper documents. Every month, more and more vendors[4] are touting a process called document imaging as a solution to ubiquitous paper-related problems.

We became interested in imaging after seeing several vendor demonstrations and decided to try duplicating imaging system functionality with off-the-shelf personal computer components. This paper describes insights obtained during a hands-on evaluation of imaging technologies and techniques.

Imaging

Document imaging is a way to get printed material into a computer. The imaging process involves transforming paper documents into computer-compatible files which can be managed and used in a distributed system environment. As awareness of document management increases, advantages associated with the concept of automated paperwork become more compelling. In fact, although envisioned benefits are extraordinarily desirable, they have been so elusive that knowledgeable administrators spend little time thinking about creating a paperless office with computers. And yet ...

Traditional document storage methods are unquestionably resource intensive and expensive. In response to these problems, discussions commonly cite space, accessibility, security and integrity as some of the issues to be resolved by imaging systems[5]. Imaging offers real possibilities for savings as areas previously used to store documents are converted to laboratories or offices. Further benefits of automated document storage can be obtained only through the shared processing made possible by local area networking. Projected savings are supposed to accrue through improved efficiency as people locate and retrieve documents faster and easier than they can with traditional systems. In addition to

[1] David O. Stephens, "What's Ahead for Records Management in the '90s?," The Office, January 1990, p. 135.

[2] David E. MacWhorter, "Image Is the Next Information Frontier," The Office, April 1990, p. 78.

[3] David T. Bogue, "Micrographics: Its Once and Future Technology," The Office, January 1990, p. 71.

[4] John A. Murphry, "Document/Image Management Systems: Their Advantages Are Not Optical Illusions," Today's Office, April 1990, p. 40.

[5] David E. MacWhorter, "Image Is the Next Information Frontier," The Office, April 1990, p. 78.

rapid retrieval, simultaneous access to the same document by different workers promises opportunities for literally revolutionizing document processing methodologies.

Imaging system disadvantages, both apparent and subtle, must be anticipated and resolved if an imaging application is to be successfully used in any organization. With complete turn-key imaging systems selling for hundreds of thousands of dollars, the most obvious problem is one of cost. Other, far more threatening difficulties include: employee resistance to change; problems typically associated with automation of manual practices and procedures; dangers inherent in over-dependence on a single vendor or on a vendor's proprietary system; and problems caused by unrealistic expectations - like the idea that document imaging will finally lead to the paperless office.

ALTERNATIVES

The problems described thus far are quite real and document imaging sounds promising but it is certainly expensive and will likely introduce new problems to be solved. So what is the most practical response?

Well, there are only two choices. Either continue to use traditional methods for storing and archiving documents (*ie.* stacks and piles, folders, cabinets, *etc.*) or implement something new. Advantages and disadvantages of alternative media like microfiche[6] are well known and while a few developments in fiche are still forthcoming, this is a mature technology which has done little to reduce dependence on paper documents. Continued use of microfiche and automated filing equipment is assured but the question to be asked is: "Are these enough to cope with increasing volumes of paper?".

Before document imaging can be used to solve paper processing problems, choices as to extent and approach remain to be made. Personal computers and a local area network may provide a reasonable alternative to commercial imaging systems. While to a certain extent, it is possible to pick and choose among vendor offerings, system components cannot be purchased like delicacies from an imaging buffet without fully understanding the ramifications of each decision.

This investigation into document imaging was intended to duplicate the functionality of commercial imaging systems with readily available PC hardware and software while avoiding major programming efforts. Adoption of these project constraints restricted our attention to issues of image capture, storage and retrieval. Complications associated with operational aspects of LANs (local area networks), groupware, and database maintenance are acknowledged but were not actively investigated.

SMALL-SCALE IMAGING

Two very distinct concepts are possible when talking about document imaging. The most common idea is to make a digitized duplicate of a document. Under this approach a scanner is used like a camera to take a "picture" of the original, saving text, line-art

[6] Steve Davis, "Micrographics is Increasing Its Exposure," Today's Office, April 1990, p. 46.

drawings and photographic figures as a single graphic file. Text imagery on the other hand, depends on optical code recognition (OCR) to convert scanned text into standard word processing documents while intentionally disregarding drawings and figures.

System requirements for image capture workstations for both methods are quite similar. Hardware configurations consist of a powerful microcomputer, scanner, laser printer, and some form of high-capacity data storage device such as WORM (Write-Once, Read-Many) optical drives. Additional imaging system requirements include a LAN for distributed access to the image database and software designed for image capture, indexing, database maintenance, and *ad hoc* selection of specified images.

The list of system hardware components employed in this exercise included IBM PS/2s, an IBM AT, a Hewlett-Packard (HP) ScanJet Plus scanner and LaserJet III printer, with an external IBM WORM drive. Evaluated software consisted of Microsoft's *Windows 3.0*, *Scanning Gallery Plus* from HP, Precision Software's *Superbase 4 Windows*, Caere's *OmniPage/386* and *askSam* from askSam Systems. Each of these products was used because they were already on-hand and mention of a specific item should not be taken as a recommendation, although with a single exception, all have been named in leading industry publications as being among the best examples of their kind.

Graphic Imagery

As mentioned, graphic document imaging can be likened to taking a photograph of a document. Everything on the original page including handwritten notes, date and time stamps, any alterations, figures, drawings and typed or printed text is saved exactly as it exists when the document is scanned. Like a photograph, once the image has been acquired few actions beyond displaying or printing the image are possible.

Scanning Gallery Plus is a *Windows*-based image capture program supplied with HP's ScanJet Plus. As this paper was being prepared, we received an upgrade to version 5.0. This version offers substantial improvements over the previous program with *Windows 3.0* compatibility, an integrated paint program, easier operation and options for saving scanned images in established graphic formats including .TIF, .PCX, and *Windows* Clipboard.

Capturing an image requires two major steps after placing a document under the scanner cover. After selecting the area to be captured from the *Preview Scan* screen representation, *Scan Region* actually captures and saves the document image as a file. Operator choices range from the type of image processing used for photographic figures to exposure characteristics and image scaling.

The principal problem related to graphic imagery is one of file size. Graphic TIFF images are large. Since every part of the page is saved regardless of the presence or absence of ink, TIFF file size varies directly with resolution and the size of the area being scanned. An image of a full page scanned at 300 dpi (dots per inch) can be expected to reach a size of a megabyte or larger. The scope of this observation becomes clearer by envisioning a 115Mb hard disk worth $2,077 (State discount price for Florida's universities) holding about 100 page images. Obviously, far too expensive for serious consideration in a production environment, this fact is a driving force behind growing interest in optical mass storage devices.

The importance of archiving graphic document images on uneditable optical media cannot be overstated but a brief warning is in order. While it is true that once saved on an optical cartridge an image cannot be easily modified, it is quite simple to alter a TIFF file using any number of graphics paint programs **before** being copied to the WORM. Administrative procedures with traditional checks and balances should be sufficient to deal with this possibility but organizations are entitled to know about prospects for unauthorized image manipulation.

Large file sizes also have deleterious effects on the amount of time needed for saving, retrieving, displaying and printing these images. All of these issues lead to system requirements for powerful microcomputers and peripherals which drives up imaging costs. Other problems are associated with the fact that the appearance and composition of a document has direct effects on scanning variables. Not all image enhancement processes offered by *Scanning Gallery Plus* are acceptable for a particular figure. Some degree of operator experimentation and experience is often needed to acquire an acceptable image which extends the time needed to complete the capture process.

Graphic imagery advantages are compelling. Relatively inexpensive hardware and software can be used for total preservation of original document appearance. The absolute value of digitized and electronically-stored images can only be inferred at this time but may actually be worth more than the original documents because they can be copied and restored for years without any possibility of appearance degradation over time. TIFF files also offer the possibility of post-capture processing by OCR software which further heightens the potential value of scanned graphic images.

TEXT IMAGERY

Text imagery is based on the premise that words are the most important aspect of a document. Optical character recognition provides a means of scanning printed text to create word processing documents. Although there are growing numbers of OCR programs capable of accurately interpreting a printed page of text, on the basis of speed and accuracy *OmniPage/386* has been a consistent winner in head-to-head evaluations of OCR software[7].

OmniPage is a page recognition program capable of working with single or multiple-column pages and successfully handles a wide variety of fonts, type styles and text sizes. Users have the option of performing the recognition process as each page is scanned or using TIFF images after they have been captured. If recognition is done as each page is

[7] Bill O'Brien, "OmniPage/386," in "OCR Software Moves Into The Mainstream," Lori Grunin, ed., PC Magazine, October 10, 1990, p. 315.

scanned, an opportunity to incorporate a number of separate pages into the final document is offered.

The process begins as *OmniPage* interprets the layout of the page, blocking out paragraphs and non-editable graphic areas, leaving a painted page image on the screen. Text recognition converts the bit-mapped image to ASCII characters and finally, a transitional editor presents text with known errors shown by tilde (~) marks for final editing. Be assured there will be errors. Appearance quality of the original (*e.g.* smudges, fingerprints, fuzzy copy, dot-matrix print) has nearly as disastrous an effect as does skewed placement or a dirty scanner glass. Other errors will be created by underlined descenders like the letters q, y, and p in the words quality and mapped, as well as outsized or otherwise unrecognized fonts[8].

Performing accurate OCR takes substantial amounts of time. Beyond problems already described, the chief difficulties of optical character recognition have to do with efforts required to locate and correct errors in the resulting text. Our work with OCR suggests nearly half of all errors go unrecognized and therefore are not indicated with tilde marks which necessitates painstaking editing. In the worst case, editing must be accomplished with the original page immediately at hand. This function can take far longer to complete than do the scanning and recognition processing steps combined.

To their credit, text files are much smaller than graphic files. The same 115Mb drive (remember - around 100 graphic page images) would hold more than 23,500 single-page (5Kb) documents. Text imagery makes it possible to use text captured from extensive printed resources without time consuming and even more error-prone retyping. Normal word processing experience provides the only skills needed to edit and use these documents.

Since text files created by OCR are no more than word processing documents, any PC able to satisfy the organization's word processing needs will work for text imagery users. Users of graphic files on the other hand, must have relatively powerful PC workstations simply to handle the load imposed by large files and graphics processing requirements.

ROLE OF DATABASE MANAGEMENT SYSTEMS

If the only purpose of the imaging process were to preserve documents, then the previous discussion of capturing graphic and text images would be complete. However, the ultimate value of imaging will be realized only when personnel and other resources are diverted from activities revolving around filing, finding and moving paperwork to activities designed to enhance extraction and use of information contained in each document.

Database management systems or databases, are programs designed for rapid retrieval of specific records on the basis of data contained in one or more fields within each record.

[8] Lori Grunin, "OCR Software Moves Into The Mainstream," PC Magazine, October 10, 1990, p. 320.

Once graphic images and OCR-processed text documents have been saved as files, a database is needed for selective retrieval of indexed data along with images.

GRAPHIC DATABASE - *SUPERBASE 4 WINDOWS*

Few database products are capable of dealing with graphic images and even fewer are able to display an image and field data simultaneously. *Superbase 4 Windows* is a programmable, *Windows 3.0* compatible, product offering relational traits and the ability to show a graphic image in a pre-set window. Operating at several levels of programming complexity, *Superbase 4* is able to reference and display external TIFF files with surprising ease.

If scanning a document page is counted as the first step in the imaging process, indexing resulting image files comprises the second step. It is possible to minimize the impact of this process by creative programming (*ie.* point and click to select a specific image file or automatically updating a field with today's date) but other information to be used as a record index must be entered manually. Compromise between system flexibility - many record fields, and entry speed - fewer fields, will be ruled by user-designed system constraints.

Graphic databases share certain disadvantages which in this case are related to graphics processing requirements making a strong microcomputer a must. Even still, it takes a fair amount of time to paint the screen with a full page image and if the number of hand-keyed data fields is limited, database search options are limited proportionately. In addition, the programming needed to ensure effective implementation infringes on our objective of avoiding major development efforts.

At the most basic level, associating graphic images with *Superbase 4* database records is quite easy. Given the claimed ability to create very large databases, *Superbase 4* may be appropriate for a number of diverse imaging applications and since a network version of the database is available (but was not evaluated), this product fulfills many of the basic needs for managing a graphic image database.

TEXT DATABASE - *ASKSAM*

askSam is a programmable, DOS-based, free-format database with limited hypertext capabilities. This product can automate the process of distinguishing between an extensive number of ASCII text files and unlike traditional database management systems, is able to locate all occurrences of specific words by employing full-text search algorithms.

Unfortunately, although *askSam* is an extremely innovative product, its disadvantages seem to preclude use as an integral component of a document imaging system. Text files must be converted to special formatting and some programming with a non-standard language is required for most effective use of *askSam*. Furthermore, while *askSam* is able to work with graphic images after a fashion, it does so by launching someone else's graphic display program from within the database. This approach precludes a simultaneous view of a graphic image and associated database information.

To it's credit, although one of very few free-format databases, *askSam* is an inexpensive application. System requirements are quite modest unless graphic display options are used and a network version (not evaluated) is available.

WORM ARCHIVING - IBM's 3363 EXTERNAL WORM DRIVE

Optical data storage is especially attractive in document imaging applications because individual optical disk cartridges can hold hundreds to thousands of megabytes of data. Daisy-chaining (interconnecting) multiple drives and optical jukeboxes (single drive with multiple cartridges) can extend this capacity to hundreds of gigabytes of on-line mass storage. On the basis of published articles, many of the following observations might be resolved simply by using a better example of this critical peripheral.

IBM's 3363 WORM is a ***slow*** device. Reports on the effects of daisy-chaining SCSI devices to achieve greater storage capacity suggest access times across chained WORM drives would decline significantly below those experienced with our configuration. Even though our drive has been installed for two years, WORM technology is still relatively immature with arguments raging about purely technical issues[9] and suffers from a general absence of acceptable device drivers. At 200Mb per single-sided cartridge, this drive has the smallest capacity of any WORM drive currently available. The industry standard for 5.25" optical media is somewhere between 500 and 600 megabytes per cartridge.

However, because storing document images on optical media can preserve the unaltered appearance of original documents for years (claims of data life expectancy on optical disks range from thirty[10] to a hundred[11] years), and because the cartridges themselves are impervious to many conditions which would easily destroy magnetic tapes or disks, WORMs are considered extremely attractive mass storage devices. The write-once characteristic of this optical technology is conducive to archiving records and creates relatively fool-proof audit trails but questions about the legal validity[12] of WORM-archived originals remain unanswered.

OBSTACLES TO SUCCESSFUL DOCUMENT IMAGING

Thus far, issues of image capture, indexing and storage have been presented. These steps, though crucial, appear to represent no more than a beginning for the document imaging process. This attempt to duplicate commercial imaging system functionality has not been successful in devising a working alternative but has provided important insights into relevant issues. While technical aspects of image capture appear to be straightforward, at

[9] David A. Harvey, "State of the Media," in "State of the Art: Magnetic vs. Optical," Jane Morrill Tazelaar, senior ed., Byte, November 1990, p. 275.

[10] David Kalstrom, "Getting Past the 'Write-Once' in WORM," IMC Journal: Publication of the International Information Management Congress, Jan./Feb. 1990, p. 16.

[11] David A. Harvey, "State of the Media," in "State of the Art: Magnetic vs. Optical," Jane Morrill Tazelaar, senior ed., Byte, November 1990, p. 275.

[12] Emily Leinfuss, "When Optical Storage Courts Danger," inset in "USAA's Image of Success," Datamation, May 15, 1990, p. 80.

least three major management issues (image acquisition ramifications, quality control and document management) deserve further comment.

Document imaging can obviously save time and money by reducing filing errors and making record retrieval faster but the impact of shifting human resources is seldom mentioned in vendor presentations. Many workers will have to be trained from the ground up and that training may have to begin with "This is a PC ...". And even after trained users are available, people responsible for finding filed documents will have to be assigned new tasks as efforts are increasingly dedicated to capturing and indexing images. These observations are especially noteworthy because they indicate the kind of managerial skills which will be needed to guide an organization through the sort of substantial changes[13,14] required for the most effective implementation of imaging technology.

Operators of image capture workstations will have to develop judgmental skills to assure image validity. They will have to make decisions about what portion of a scanned image to save and will have to make sure variables employed during the scan result in an accurate copy. Naturally, anything that requires this sort of attention will take more time per execution than would a mass production approach.

High volume or production imaging systems are dependent on rapid mass scanning and retrieval capabilities. Our system took an average of four minutes to complete graphic image captures with HP's *Scanning Gallery Plus* (4.0). Text imaging wasn't any faster since four to five minutes were needed to scan and interpret each page. However, unlike *Scanning Gallery Plus*, *OmniPage/386* can handle multiple page documents with the aid of HP's automatic document feeder which at least offers the promise of batch scanning possibilities.

Before any database can be used with reliance, users must have full confidence in the validity of all system data. Regardless of the document imaging approach (graphic or OCR text), inaccuracies caused by scanning problems or document variations makes it seem reasonable therefore to suggest that operators MUST proof each image and edit OCR documents **before** files are finally committed to the document imaging system. Yet another instance of time-consuming activity seldom mentioned in imaging system demonstrations.

Then there is the question of what to do with existing stockpiles of paper records. There are arguments in favor of scanning everything at once or scanning nothing but new documents but the most practical compromise might be to scan everything new and catch up with archived documents as circumstances and resources allow.

A recent review of document management software described systems designed to manage text (and/or images) which have been scanned from paper documents and saved as cross-

[13] Roger E. Wasson, "Organizing for Future Technologies," Datamation, April 1, 1990, p. 93.

[14] Gary H. Cox, "Technology's Rewards Without the Risks," Datamation, February 1, 1990, p. 69.

referenced computer files[15]. While we were not able to review any examples of this new class of software, their reported capabilities seem to address the most important deficiencies of our database investigation. *Marvin 3.0* from ImageTech, Inc. may be worth serious consideration because it is *Windows*-based but with a list price of $9,500 to $33,500 for a multiuser version[16] it is not a trivial purchase and will have to be subjected to an extensive hands-on evaluation before adoption.

SUMMARY

Our strategy for home-grown microcomputer imaging suffers from at least three critical flaws. Because *Scanning Gallery Plus* offers no option for batch processing, no capability for volume processing of graphic images exists. Although another vendor's product would probably be satisfactory, our WORM disk drive is too slow and too small to be of any significant value in a high volume application like this. And finally, because our databases do not offer any particular ability to manage the proliferation of scanned images and OCR documents expected in an imaging environment without special programming, consideration has to be extended to specific multiuser document management software.

Document and image management is clearly the key concept for successful imaging applications. As noted, several deficiencies of our off-the-shelf components can be offset by using better products and still others may be resolved by technological innovations within another year or so. Organizations considering adoption of imaging as a strategic application would be well advised to establish a monitoring program to watch improvements in mass storage technology and document management programs.

In the meantime, the products used in this exercise are perfectly adequate for project-level applications. Our office will continue to rely on functions provided by this hardware and software to develop innovative solutions as needs arise. Document imaging is unquestionably important enough for substantial investment as long as proprietary or single-vendor systems can be avoided.

Institutions suffering from the paperwork processing problems outlined in this discourse stand a good chance of becoming involved with imaging within the next few years. Despite vendor hype, innovations such as document imaging must explicitly address problems in your organization before they can be considered as solutions. A judicious mixture of programmed flexibility and calculated response is helpful in any introduction of advanced technology, however the ultimate secret for a successful implementation of new technologies will always be to balance organizational expectations and needs against available resources. Good luck.

[15] Chet Schuyler, "Document Managers Bring Law and Order," PC Week, October 1, 1990, p. 93.

[16] Op. Cit., p. 94.

ACKNOWLEDGEMENTS

Grateful appreciation is extended to Scott Farmer who assisted in the evaluation of *askSam* and *Superbase 4 Windows* and to Edward Gilbert and Mack Casey who suffered through my practice sessions with such admirable tenacity.

REFERENCES

Bogue, David T. "Micrographics: Its Once and Future Technology." The Office, January 1990, pp. 71.

Cox, Gary H. "Technology's Rewards Without the Risks." Datamation, February 1, 1990, pp. 69-73.

Davis, Steve. "Micrographics Is Increasing Its Exposure." Today's Office, April 1990, pp. 46-47.

Grunin, Lori. "OCR Software Moves Into The Mainstream," PC Magazine, October 10, 1990, pp. 299-356.

Harvey, David A. "State of the Media" in "State of the Art: Magnetic vs. Optical." Jane Morrill Tazelaar, senior ed., Byte, November 1990, pp. 272-339.

Kalstrom, David. "Getting Past the 'Write-Once' in WORM." IMC Journal: Publication of the International Information Management Congress, Jan./Feb. 1990, Vol. 26, No. 1, pp 16-17.

Leinfuss, Emily. "When Optical Storage Courts Danger." inset in "USAA's Image of Success", Datamation, May 15, 1990, pp. 77-82.

MacWhorter, David E. "Image Is The Next Information Frontier." The Office, April 1990, pp. 78.

Murphry, John A. "Document/Image Management Systems: Their Advantages Are Not Optical Illusions." Today's Office, April 1990, pp. 36-40.

O'Brien, Bill. "OmniPage/386" in "OCR Software Moves Into The Mainstream," Lori Grunin, ed., PC Magazine, October 10, 1990, pp. 299-356.

Schuyler, Chet. "Document Managers Bring Law and Order." PC Week, October 1, 1990, pp. 93-97

Stephens, David. O. "What's Ahead for Records Management in the '90s?" The Office, January 1990, pp. 135-136.

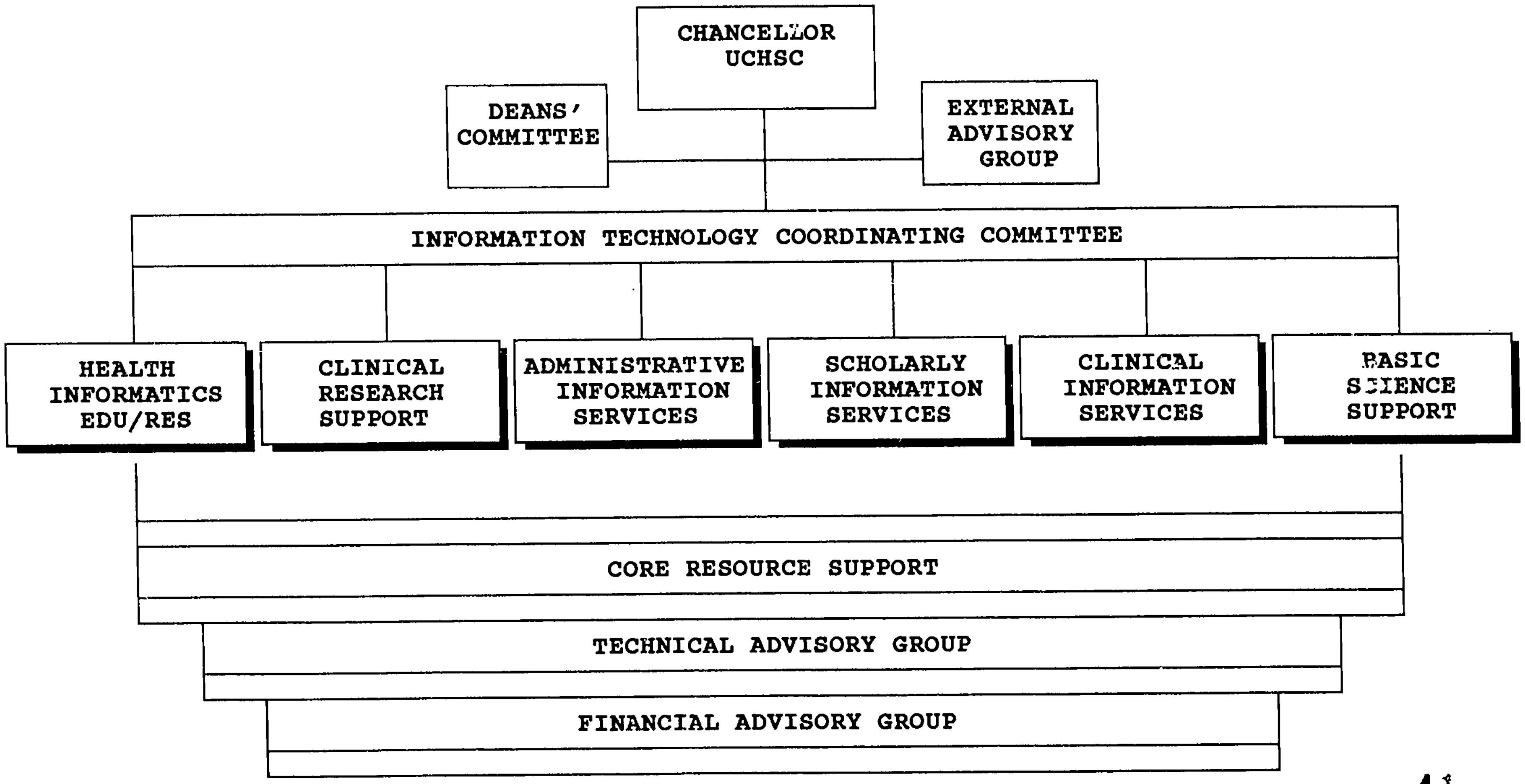
Wasson, Roger E. "Organizing for Future Technologies." Datamation, April 1, 1990, pp. 93-95.

CLIENT-CENTERED STRATEGIC PLANNING

Charles Bandy
University of Colorado Health Sciences Center
Denver, CO

The University of Colorado Health Sciences Center is beginning a client-centered strategic planning process to relate investments in information technology to the support of campus goals. User groups are being established for the major areas of campus activity. These groups will define present and future needs related to their area of specialty and interest. A coordinating committee will compile these results and review existing strategic plans for each of the schools and for the campus administration. The coordinating committee and service units will collaborate to produce the strategic plan for information technology for the campus. The goal of the process is to provide integrated client-centered planning and a structure for implementation.

The model is based on a planning structure at Columbia Medical Center in New York. Both have a large number of relatively autonomous units and the hospitals are separate corporations, requiring intense interaction and cooperation to accomplish the goals of each.


CHANCELLOR UCHSC
DEANS' COMMITTEE
EXTERNAL ADVISORY GROUP
INFORMATION TECHNOLOGY COORDINATING COMMITTEE
HEALTH INFORMATICS EDU/RES
CLINICAL RESEARCH SUPPORT
ADMINISTRATIVE INFORMATION SERVICES
SCHOLARLY INFORMATION SERVICES
CLINICAL INFORMATION SERVICES
BASIC SCIENCE SUPPORT
CORE RESOURCE SUPPORT
TECHNICAL ADVISORY GROUP
FINANCIAL ADVISORY GROUP

INTRODUCTION

The University of Colorado Health Sciences Center in Denver is implementing a planning process that is client-centered. User groups are being established for all major areas of on computing and other types of information technology. These groups will define present and future needs related to their area of specialty and interest. The goal is to provide an integrated client-centered planning effort and a structure for implementation.

The Health Sciences Center is one of four campuses of the University of Colorado. Each campus has a Chancellor, who reports to the President. There are two vice chancellors on campus. One, though not having the title, functions as Executive Vice Chancellor. The second is the Vice Chancellor for Academic Affairs. The Associate Vice Chancellor for Information Systems reports to the Executive Vice Chancellor. Information Systems includes Administrative Data Processing, Telecommunications, the Computer Information Center, Data Communications and Bioengineering.

There are schools on campus: the schools of Dentistry, Medicine, Nursing, Pharmacy, and the Graduate School. A number of Allied Health degree-granting programs are also on campus. Unlike general academic campuses, the Deans do not report to the Vice Chancellor for Academic Affairs. They report directly to the Chancellor. Academic health science centers have been described as a "loose confederation of power." This decentralization of power is even more evident in the School of Medicine, where department Chairmen actually have control over more funds than the Deans. This is primarily due to the fact that only 19% of the total campus budget comes from the State. The remaining 81% is from grants, contracts, gifts and revenue-producing activities. There is a constant tension regarding who has control of funds, especially those produced by indirect cost recovery (ICR). This was intensified recently by a "white paper," authored by the faculty, and calling for a greater per centage of ICR monies to be returned directly to the department producing them.

This is very pertinent to our planning effort, as I will point out later.

Rationale for a Client-Centered Approach to Planning

The most compelling reason for a client-centered approach to planning for information technology is that it emphasizes **information**, rather than technology. Secondly, investments in technology are determined by actual and projected needs of the clientele being served.

> "Management must determine whether the recommended changes are simply to satisfy the desires of purely technically oriented data processing individuals, who often are more interested in implementing the latest technology than satisfying the information needs of end users, **or** to satisfy the needs of end users, who are concerned less with technology than simply getting information when they need it in a form that is useful."[1]

On the very practical side the clients provide funding, either directly or indirectly. Consequently, staying close to their needs provides a much broader base for support.

PROGRESS ON PLANNING FOR INTEGRATED INFORMATION MANAGEMENT

For almost a decade several academic health sciences centers across the country have committed themselves to implementing integrated academic information management systems (IAIMS). Planning at the University of Colorado Health Sciences Center is done in this context, although it is a goal for us and not currently implemented. IAIMS is a concept that comes from a report funded by the National Library of Medicine (NLM) in conjunction with the Association of American Medical Colleges (AAMC). Published in the 1982 October issue of the Journal of Medical Education,[2] the report laid the groundwork for planning for the use of information technology in academic health sciences centers.

Originally, the NLM selected four academic health sciences centers in 1983 to investigate the IAIMS concept and to develop prototypes. Currently, there are eleven institutions receiving funding from NLM for planning, modeling and implementation.

The thrust of the IAIMS initiative is to provide a coherent plan for the development of campus computer and telecommunications networks and services, which will provide faculty, staff and students with efficient access to relevant databases (e.g. library files, patient information, research files, remote databases such as MEDLINE, etc.), and provide the institution with a significant strategic tool to assist in achieving its objectives. Separate, decentralized computer operations are recognized and minimal control is exercised over processing. IAIMS maximizes control over networking. This assures the necessary blend of autonomy and coordination.

Benchmarks of Information Technology Activities

All four campuses are now connected with fiber optic. Almost all buildings on the Health Sciences Center campus are linked with fiber.

In 1986 the Chancellor appointed a Computing Advisory Board to advise him on planning and expenditures for computing. Through various sub-committees, the Computing Advisory Board was instrumental in preparing network protocol standards, authoring a document to establish the Computer Information Center, selecting electronic mail software, and sponsoring a Medical Informatics Seminar Series. One of the first projects was to prepare a manual on ergonomics for managers and purchasing agents. One of the last accomplishments was to recommend the client-centered planning approach to the Chancellor.

In 1988 a planning grant was submitted to the National Library of Medicine. This was approved, but not funded. The grant preparation process did, however, bring together a nucleus of individuals who have continued to work towards integrated planning.

In 1989 the Chancellor created the position of Associate Vice Chancellor for Information Systems. Originally charged with total campus computing service, for a number of reasons, it is now limited to administrative computing.

Two events in 1990 will have a significant impact on the computing environment at the Health Sciences Center. The Financial Reporting System was outsourced to the University's central computing system in Boulder. Also, the University Hospital was converted to a private, not-for-profit institution with its own President and Board. The Hospital is now in the process of hiring its own Chief Information Officer and transferring computer operations. This is of major concern, since the campus mainframe is currently operating at only 33% capacity, including the hospital applications.

DESCRIPTION OF PLANNING PROCESS

In the Spring of 1990 the Computing Advisory Board charged me with preparing a draft of a strategic plan for information technology on campus. In May I submitted a concept proposal, recommending that we use a modified model of the program at Columbia-Presbyterian Medical Center in New York. The campuses are similar in that they both have a large number of relatively autonomous units and in both cases, the hospital is a separate corporation, requiring intense interaction and cooperation to accomplish the goals of each.

The Information Technology Coordinating Committee is the overall steering committee and its Chair reports directly to the Chancellor. Feeding information to the Coordinating Committee are representatives from the client groups, the key feature of the model. There are six client groups: Health Informatics Education/Research, Clinical Information Systems, Scholarly Information Systems, Clinical Research Services, Basic Science Support, and Administrative Information Systems.

Health Informatics Education/Research. This group is concerned with the use of computers in health science education, information and computer literacy, and the broad applications of computing in the practice of medicine.

Clinical Research Services. This client group is made up of individuals interested in downloading subsets of data from the local hospital information system for use in clinical research.

Scholarly Information Systems. The main focus of this group are the information systems provided and planned by the Library, including network access to an online catalog, and a gateway to a third-party vendor for twenty-four hour per day access to the MEDLINE database. It includes access to full-text databases.

Clinical Information Systems. The chief emphasis of this group is the development of the hospital information system at University Hospital.

Basic Science Support Services. Needs of this group of faculty and students are primarily related to scientific computing, although fairly extensive database management capability is required.

Administrative Information Systems. In addition to the traditional administrative systems (e.g. personnel, payroll and purchasing), this group includes departmental administrative and school-specific applications.

Information Technology Coordinating Committee. This Committee has primary responsibility for planning and implementation, keeps the Chancellor, Vice Chancellors and Deans informed of plans and secures their approval. Heads of service units are members of this Committee, as well as a representative from each of the User Groups.

Technical Advisory Group. The Technical Advisory Group consists of members of the former Computing Advisory Board. When this planning process was initiated, it took the place of the Computing Advisory Board. It provides technical expertise to each of the User Groups and to the Information Technology Coordinating Committee. A member of the Technical Advisory Group is assigned to a client group. Each client group has a representative on the Information Technology Coordinating Committee.

Core Resource Support. This group is comprised of representatives from each of the institutional support units that provide computing and related resources and services. Telecommunications, Instructional Computing and the Computer Information Center are examples.

Financial Advisory Group. The Financial Advisory Group will provide fiscal planning expertise to all entities in the planning process, primarily to the Information Technology Coordinating Committee. It will be composed of a representative from each of the Deans' offices and an individual from the Chancellor's Budget Office.

PRESENTING THE PLAN TO THE CAMPUS

Following acceptance by the Computing Advisory Board, the concept was presented to the Chancellor for his approval. He appointed me as of Chair of the Coordinating Committee. At that point we compiled a list of faculty members and others for "direct marketing" in each of the schools. Brief fifteen-minute presentations were made in the offices of these selected faculty to members explain the concept, elicit their support and to gauge the degree of enthusiasm we might expect before approaching the Department Chairs and the Deans. Next we met individually with the heads of the various computing clusters on campus, in some cases LANS and in others minicomputers.

Only after we had contacted important users and computing centers did we meet with the Deans and Vice Chancellors, again on an individual basis. Now we were ready to present to the bi-weekly meeting of the Chancellor's staff, Deans and Vice Chancellors. The groundwork paid off. Already having the individual commitment, the approval of the group was assured.

The School of Nursing, due to the leadership of their Director of the Center for Nursing Research, had already incorporated planning for information technology in their School's strategic plan, completed in October 1990.

The School of Medicine has strategic planning groups for seven different areas. A presentation was made to the chairs of these groups and members of the information technology planning team will be working with each of the seven groups to assess the need for information technology support.

The Faculty Senate of the School of Medicine has recently made the decision to form a special planning group devoted exclusively to strategic planning for information technology.

Meetings are scheduled in December with faculty of the Schools of Dentistry and Pharmacy. Following these sessions, focus groups for each of the six client groups will begin.

STRUCTURE FOR IMPLEMENTATION AND FUTURE PLANNING

Recently, the Committee for Academic Programs and Planning issued a report containing a strong recommendation for the Office of Academic Affairs to assume responsibility for planning for information technology. Presently, the position of Vice Chancellor for Academic Affairs is vacant. How this will affect the timing of our planning is still unknown.

One of most important issues we will be addressing is the relationship between the Department of Information Systems and the computing needs of the academic community. Units of Information Systems (e.g. Telecommunications) provide part of the Core Resource

Support. However, traditionally, they have not been charged to serve the needs of the Schools. Rather, they have concentrated almost solely on administrative systems.

Once our plan is complete, and representative of clients needs, "...management must ... be prepared to make the necessary capital investment in true integration technology. The investment must be viewed as a contribution toward achieving the long-rang strategies and goals of the entire organization."[3]

CONCLUSION

A longer time period is required for planning when a client-centered approach is used. All participants should be aware of the balance required between expressed client needs and the necessity of time-constrained management decisions. For the long term, greater involvement of clients in the planning process increases the chances of continuation of program. Addressing specific needs of clients facilitates the documentation of the success of planning and implementation.

Client-centered planning will help a campus to maximize its investment in information technology, ensure integrity and availability of data, and relate planning and expenditures to activities directly supporting campus. It provides a setting for truly integrated planning.

[1] Lemon, Robert and Crudele, John. "Systems Integration: Tying it all Together." Healthcare Financial Management, June 1987, p. 53.
[2] Matheson, Nina. "Academic Information in the Academic Health Sciences Center." Journal of Medical Education, October 1982, v. 57, no. 10, pt. 2, pp. 1-120.
[3] Lemon, Op cit, p. 53.

A Degree Audit System Implementation in a Distributed Environment

Bruce L. Rose
SCT Corporation
Cleveland, Ohio

James Armstrong
Cuyahoga Community College
Cleveland, Ohio

Vitaly Zavadsky
SCT Corporation
Cleveland, Ohio

Abstract

Cuyahoga Community College (CCC) and Systems and Computer Technology (SCT) have developed and implemented an Advising and Graduation Information System (AGIS). This information system centers around a powerful PC-based degree audit application, which relies on background micro-mainframe communications to retrieve both student records (academic history or transcript) and degree/catalog information from "authority" databases.

Because of the PC architecture and network scheme, the AGIS application is highly interactive - allowing views and browses of the authority degree catalog, realtime audits during counseling sessions, and "degree shopping" inquiries. The second phase of the project will deliver a batch or "distance advising" function which will distribute automated degree audit processing and mailing capability to strategic LAN based servers and laser printers throughout the student and academic support areas of the College. The highly portable nature of the application will allow for the possibility of . widespread automated faculty advising.

Presented at Cause 90
Miami, Florida
November, 1990

Introduction

Similar to other papers found in the Cause 90 Proceedings, the authors wish to tell a story. The story is about the development and implementation of a software application package called AGIS (Advising and Graduation Information System). AGIS is primarily a degree audit system; but, as the name implies, AGIS is also an information system targeted for users in the areas of counseling, admissions and records, and potentially all faculty involved in student advising.

AGIS was developed at Cuyahoga Community College, a two-year institution serving the Greater Cleveland metropolitan area. Cuyahoga Community College is composed of three campuses and a district office serving a diverse student population, with typical fall term enrollments near 25,000. The College has a long-standing partnership with Systems and Computer Technology Corporation; it is through this resource management partnership that AGIS, as well as many other information, network, and data systems initiatives have been devised and developed.

The balance of this story will describe the AGIS development effort at Cuyahoga, and the outcomes we have experienced. We will describe our own efforts against the background and history of degree audit systems, identify our own development strategies, describe the components of the AGIS system, and finally end with a summary of our achievements along with a laundry list of future goals or milestones in the AGIS project.

Background

Winston and Endler (1984) have provided a concise discussion of automated advising systems found in higher education. The first computerized attempts at matching a student's course history with a corresponding educational objective or degree program occurred in 1968 at the University of California at Berkeley. Very soon after this Purdue University began to produce rudimentary Academic Progress Reports from the computer. In the mid 1970's both Brigham Young and Georgia State universities had built comprehensive mainframe applications which produced degree audit and advising reports to serve a large portion of their undergraduate communities (c.f. Spencer et. al., 1983).

Both the Brigham Young Advising by Computer (ABC) and the Georgia State Programmed Academic Curriculum Evaluation (PACE) system were notable in that they defined a complete language or table design into which the college's catalog of graduation requirements and degree programs could be systematically coded anJ maintained. This resulted in a structured electronic version of an authority file which:

- provided a machine-readable set of degree requirements establishing the basis for computerized evaluation against individual coursework, and
- provided an external data structure which could then be modified or maintained by an administrator trained in the usage and meaning of the electronic catalog language.

By 1983 an AACRAO survey noted that 132 colleges and universities had some form of degree audit or advising systems in place or under development. In 1984, Brigham Young sponsored a conference on Degree Audit systems (Peterson, 1984) which was attended by Cuyahoga representatives. Degree Audit systems from a number of schools were presented and reviewed in terms of functionality, cost/benefits, ease-of-use, and other factors.

Each system was similar, especially in the following respects:

- Nearly all were mainframe based in keeping with the expected volume and size of the application, batch requirements, security, and concurrent access needs.

- Most systems began their "lives" as batch systems, producing audits en masse. As terminals and networks became more pervasive and costs for computer cycles spiraled down, developers began retro-fitting or rewriting their applications to support on-line activities.

- Most systems had long development lifecycles and underwent significant tuning and enhancements. Acceptance and "burn-in" were often painstaking, and tended to lengthen the development lifecycle.

Not surprisingly, methods and designs were vastly different from system to system. The following contrasts are evident from one of these degree audit conferences:

- the scope of each system was quite different. The core degree audit function and some form of individualized report were common (although report formats differed widely), but additional functionality in "degree-shopping", aggregate or statistical reports and information, on-line capabilities, and "spin-off" functions (e.g. prerequisite handling) were widely different.

- Cost to develop each system varied widely. Some schools stated very low costs considering the rigorous algorithms and large scope of the application. One (honest) developer stated that the cost was much higher than anyone anticipated.

Today, the use of degree audit or computer assisted advising is widespread in higher education. An internal study (SCT, 1990) indicates three quarters of the leading educational software vendors advertise a degree audit function as part of their student system packages. Systems such as PACE, and most notably the Miami University System called DARS (c.f. Southard, 1989) are used at many other schools. The DARS system developed in PL/I and ported to ANSI COBOL is particularly attractive because of its self-contained or "black-box" nature. DARS operates exclusively in core or working storage, encouraging the local developers to build their own drivers and reporting framework around the DARS degree audit engine.

Design Methodology

AGIS is normally configured as a cooperative processing application that draws on a host computer or database server student information system (in our case ISIS) to download current and historical student transcript information to a workstation based degree audit application. This is accomplished in realtime through a background communications link that is transparent to the enduser.

The student records are then compared against a selection of program requirements keyed to the program name and catalog year. The source of these requirements is typically drawn from a database (Btrieve) resident on a local area network server (Ethernet running Novell Netware) which holds the full compliment of degree programs from 1980 until present. The electronic catalog is more fully discussed in the next section.

The following table (Table 1.0) briefly illustrates our three-tiered approach to AGIS.

Table 1.0

Mainframe/Host	LAN Server	Workstation
Authority Degree database including Structured, non-structured, conversion table, and attribute tables.	Mirror Degree database & tables	(Mirror Degree database & tables)
Student Records	User Selected Student Records)	(User Selected Student Records)
	Application (EXE), Utilities, DBMS, and Communication drivers	(Applications)
	System Parameters	User Parameter files

The authority degree files containing the Degree Coding Language (DECOL) statements of program requirements, and other parameterizations are stored on a Bull-Honeywell mainframe. However, the user accesses an operational degree database which mirrors the mainframe files. Utilities provide the means to download all or a portion of the mainframe file to the LAN server, or in some instances to a workstation that is only connected to the mainframe.

The actual AGIS application is launched from the network server (or stand-alone). The user will normally retrieve or download an individual student's record from the host, but they may also retrieve batches of students. In addition, it is possible to store student records on the LAN or PC for later processing.

In the case of a counseling office, an individual counselor will know their day's appointment schedule through a related system called the Counseling Reservation System. They may choose to retrieve all students expected for appointments and store these locally for later perusal and auditing.

Although we do have some (three out of 66) users without LAN capabilities, the three tiered Mainframe-LAN-Workstation connection is much preferred. With a large installed user base (> 50) a reasonable amount of control and management of software releases, security, and the integrity of the degree files is only possible with the LAN server capabilities in place.

Development Methodology

At Cuyahoga, the concept of an automated degree audit was explored in detail as early as 1978. The conversion to a new on-line system placed AGIS on the back burner until 1984, when a detailed analysis was presetned on the need for AGIS at the College. At this same time, Cuyahoga developed a comprehensive technology strategy for networking via a LAN/WAN topology built around the microcomputer as an intelligent workstation.

By 1986, a new set of workstation-based or cooperative applications was envisioned as a path to mainframe independence and distributed information systems. As the Cuyahoga network expanded and matured, serious work on AGIS functional specifications were completed in early 1988.

Most critically, the decision to purchase and customize an existing audit system was decided against in favor of a "ground zero" development effort. This was made in light of the strategic network decisions, and the very good fit between the AGIS specifications and a workstation-based application. In addition, we estimated a major customization effort would be necessary to bring an off the shelf package to production in our environment. AGIS was seen as a research and development effort acting as the "testbed" for future cooperative or client-server based applications which would operate in the distributed network.

We subdivided the development effort into two major phases:

- Phase I - Within six months (beginning January, 1989), provide an interactive function for both advisors and graduation personnel to produce degree audits on-line.
- Phase II - By January 1991, deliver a batch component for "distance advising"; i.e. mass degree audit processing and mailings.

Work began in early 1989 on the coding of the system. We devised a development team consisting of computer center analysts and a specific group of users from the counseling and records area. We coined the phrase "User/Tester" to describe their role as the individuals who would mold and shape AGIS into its final form. This prototyping or iterative venture began with the creation and review of an expectations document by the development team. The following excerpt indicates some of these expectations:

> "..Your role in the development of this tool is twofold:
>
> Test the current version of the application for reliability, accuracy, and robustness (lack of errors).
>
> Review the current version in terms of functionality and suitability; i.e. does it perform its function in an efficient and understandable way. Does it fulfill its purpose.
>
> Our intent in planning the development and implementation of [AGIS] is to deliver an initial release of the software as soon as possible, so that we may receive Test and Review feedback early in the development cycle. We call this an iterative development lifecycle which has the following advantages in terms of methodology:
>
> o Unburdens the designers in that they do not have to "think of everything" in the initial design. This includes both the technical and functional (user) parts of the design plan.
>
> o Defines an interactive methodology which give a positive feedback loop whereby short-interval testing and review feedback can be incorporated into the next Iteration of the software or application and the subsequent testing and review continues."

Later on we caution the User/Testers:

> "You should expect two things from the initial release of the Degree Audit software; quite simply:
>
> It will have limited functionality
>
> It may be unstable in terms of reliability"

In retrospect, the review and acceptance of the expectations document was a vital exercise. As a group, we agreed on a clearly stated starting point where user involvement was an integral part of the development lifecycle. The development team worked through six monthly iterations, before the interactive system began to crystallize. We noted the following advantages to this method:

- A continuing and strong sense of movement and urgency to the project.
- Keeping the technical development in synchronization with the needs of the users.

- Almost immediate buy-in and enthusiasm from the small group of six User/Testers; they quickly realized the difference between Steering or Task Force committee participation and hands-on involvement in the project.

- Reaching deadlines and due dates became second nature. The monthly iterative cycle contained a series of milestones culminating in a month-end meeting of the group. This cyclic repetition became ingrained.

Additionally, the development group became close-knit. Very candid discussions took place regarding troublesome policies and operating procedures that impacted the counseling and the admissions/records organizations. These "debates" were invaluable to the on-going process, but they could quickly become diversions to the task at hand. We can also list the following negatives, which are mostly by-products of this iterative team development methodology:

- Maintaining and supporting the early release software was labor intensive. Normally, such an immature system would not be available for user prodding and poking.

- Identifying the final solution. We converged towards the final solution, but it was a matter of opinion when and in what form this would actually take place.

- Predicting total technical resources was inherently difficult. Our blueprint was a set of functional specifications and an overall system design. Critical paths, sizing considerations, and system integration concerns were difficult to anticipate and were often a moving target.

Progress on the audit function and reporting formats continued through each stage of the iterative cycle. The process also aided in developing a clear picture of the desirable style for the "electronic catalog" of degree requirements against which the student record would be audited to produce the AGIS reports. The catalog component of the system was also refined several times through the iterative development cycle.

The Electronic Catalog

Cuyahoga Community College offers six major degrees:

Associate of Arts
Associate of Science
Associate of Applied Business (25 majors)
Associate of Applied Science (40 majors)
Associate of Labor Studies
Associate of Technical Studies

A variety of non-degree programs is also available. Approximately 90 separate programs make up the list of curriculum offerings for students.

Curriculum changes are common. In some cases, changes affect course content while the list of required courses remains the same. Frequently, course numbers and/or titles are revised along with content. Occasionally, major revisions of degree requirements take place. The result is that an extensive set of curriculum documents and catalogs is needed by the College's advising personnel and graduation clerks. Many of the College's students pursue studies on a part-time basis, and interruptions in their attendance are common. Four or five years often pass prior to graduation, with attendant revisions in program requirements. Students themselves rarely have the complete sets of catalogs and course revisions to accurately follow their intended program through to completion. Under these conditions, the advantages of a concise electronic file of program requirements are obvious.

The curriculum approval process at CCC is managed through a centralized office that works with the governance committee on Curriculum, Degree Requirements, and Academic Calendar. After approval by the College's Board of Trustees, program requirements are disseminated through the Curriculum Office, which is thus the central authority site for official statements of program requirements. These are published in the College catalog; changes that take place between the publication of catalogs are announced through notices in quarterly class schedule booklets and through academic advisors.

Prior to the development of the electronic catalog files for AGIS, advisors and graduation clerks worked with the printed catalogs, supplemented by copies of memos or course change information supplied through the Curriculum Office. Especially when working with students whose requirements were governed by a catalog more than three or four years old, this process was time-consuming and had a significant potential for errors due to course numbering changes. Since access to complete information was mainly through the College's counselors, the system did not foster independent action for students. Meanwhile, professional staff spent significant time on the mundane tasks of looking up course changes and reviewing revised requirements with students, rather than assisting students with issues more likely to call for professional expertise, such as career exploration, transfer planning, or personal adjustment.

In establishing the electronic catalogs for AGIS, the College was guided by the specifications assembled by a task force of representatives from counseling, admissions & records, faculty, and the computer center. These included:

- statement of program requirements in simple English as well as in coded form,

- specification of requirements in a wide variety of forms (e.g., a specific course or set of courses, one or more courses chosen from a set, a specified number of credit hours from a set),

- capability for the use of "building blocks" of requirements that could be linked into degrees,

- use of non-course data, such as grade-point average in a course or set, total credits completed in residence at the College, test scores, etc.,

- accommodation for substitution of one course for another and for waiver of a course,

- the capability to allow a course to apply to more than one requirement, or to limit use to only one requirement,

- the use of "shorthand notation" to express ranges of course numbers or course levels, rather than listing every possible course number.

In preparing the electronic catalog, the computer center staff developed a degree coding language (DECOL) that would be used to describe a degree or non-degree program. These DECOL requirement statements are compared to the student's record by the microcomputer; various output reports can then display requirements still to be completed, or show the detailed list of requirements and how the data elements were applied to those requirements.

The electronic catalogs are referred to as the authority database. To develop this database, a file was prepared for each degree or non-degree program. Since the College's on-line student record system was implemented during the 1979-80 academic year, it was decided to build the authority database with every program offered from the 1980 academic year forward. The source of official information on each program was the Curriculum Office.

Each program file consists of two sections: 1) an English-language listing of requirements, including course numbers and titles and certain narrative statements; and 2) a series of DECOL statements which express the requirements in structured, machine-readable form. Courses are listed by numbers currently available; a component of the AGIS program matches discontinued numbers from a student record to the currently available equivalent. The English narrative requirements point out such number changes, e.g., "MATH 116 - Technical Mathematics (formerly MATH 103)." The initial creation of the dozens of files for all catalog years back to 1980 occurred during 1989, and involved significant contributions from clerical staff (text entry), counselors (proofreading, validating, and correcting), and the Curriculum Office (resolving questions concerning College-wide standardization of course revisions).

As mentioned earlier, AGIS is a microcomputer-based system. For security, accuracy, and consistency, the authority database of program requirements is stored on the College's mainframe computer. Programs can be downloaded for use in academic advising or graduation checks. Most AGIS users have access to a Local Area Network (LAN) version of the system, so management of the electronic catalogs for the LANs can be handled by one person. The College has identified one of the counselors to act as the database manager.

Clerical staff assist with text editing on the degree files and with uploading of new and revised curriculum to the mainframe. The counselor downloads all updated programs for LAN users, and notifies them concerning changes via electronic mail. Non-LAN users are notified by memo in the campus mail system, and each can then download any new programs, as needed. Questions or problems regarding curriculum and requirements are communicated to the counselor for review with technical staff, the Curriculum Office, or other personnel, as appropriate.

As AGIS moved through the development cycle, the DECOL language has been improved. DECOL I was limited in expression of courses as either individual course numbers or as certain wildcards which indicated that any course in a given department could be applied. DECOL II introduced shorthand symbols which allowed for numeric ranges of courses to be expressed more simply; further, the courses within the range could be linked by "AND" or by "OR" conditions through the use of simple symbols. The development of attribute tables allowed varied groups of courses in different departments to be represented by a single coded expression in the DECOL requirement statement, e.g., a single symbol for laboratory science courses directs the matching logic to a table listing all possible departments and course numbers which fulfill this requirement. Thus, the simple English statement, "complete any college-level laboratory science course" is represented in DECOL by a single expression, rather than by a lengthy list of possible departments and courses. Some examples of the non-structured narrative statements and the structured DECOL follow:

Minimum competence in Communications by completion of the following:

ENG 101 - College Composition
ENG 102 - College Composition
ENG 103 - College Composition

[ENG 101->103]

Minimum competence in natural sciences by completion of any college-level laboratory science course

[$LAB *]

Completion of ONE of the following courses:

ART 101 - Art Appreciation
MUS 103 - Survey and Appreciation of Music
THEA 101 - Theatre Appreciation
Any 200- level English course

[ART 101 or MUS 103 or THEA 101 or ENG 2??]

Completion of a minimum of 6 credit hours selected from any combination of the following courses:

ECON 161 - Principles of Economics
ECON 162 - Principles of Economics
Any Political Science course(s)
Any 200-level Sociology course(s)

[@ 6 ch @ (ECON 161-162 or 2xPOL * or 2xSOC 2??)]

Lists of such statements can be combined to describe all the elements of a degree program. Standard "building blocks" of common degree requirements can also be utilized and combined to create degree programs. Future planning calls for a method to generate English-language text statements directly from DECOL statements. Such an innovation will eliminate the current possibility that a single program file could have a disparity between the English and DECOL description of the program requirements.

The development of the electronic catalog/authority database brought out several major organizational issues at the College. It was found that advisors or graduation clerks at different campus locations sometimes had different approaches to determining requirements where there had been significant curriculum change over the years. The need for precise definition of degree requirements (clear enough for a "dumb machine" to understand) brought out that certain programs contained requirements which were open to multiple interpretations; again, different department heads had developed different practices at different times or campus locations.

The curriculum review undertaken in preparing the authority database helped to identify the issues so that academic authorities could achieve uniform resolution for all campuses. Built into the centralized database, the best thinking of College-wide authorities becomes accessible to all users. Thus, AGIS offered a management tool to help achieve consistency throughout the College. Further, the Curriculum Office now reviews program requirements more closely to assure precision and clarity of meaning to avoid future possibilities of differing interpretations.

Curriculum change had been gradual, but AGIS allowed for ready comparisons of a specific program as it existed in 1980 and as it looks today. This brought greater awareness of the extent to which some programs had changed and raised a new issue for the College: the question of how long a student might be entitled to pursue the original curriculum, and at what point the College might require a student to fulfill more current graduation requirements. As a fairly young institution, CCC had not set out rules to advise students on which catalog they would need to follow. Policy/procedure development for "catalog in force" is now underway.

Summary

In summary, the electronic catalog/authority database is at the heart of the College's directions for academic advising and automated records processing. The process of implementing this system has also provided opportunities for organizational improvements, consistency of standards, and the creation of new management tools for the College's curriculum process.

Counselors currently use AGIS reports in academic advising, and the records offices are utilizing the system to notify potential graduates of remaining requirements. Individualized AGIS reports, in a letter format, were sent to all students who petitioned to graduate as of the end of Winter Quarter 1991; these mailings were timed to arrive prior to the beginning of the registration period, so potential graduates could use the AGIS report to select their final courses. Pilot mailings of batch advising reports are planned for mid-1991, when students in selected majors will be mailed an AGIS report as a quarterly update on their progress in the program.

Degree Audit capability is becoming an increasingly basic student service, as well as a system platform for curriculum analysis, scheduling and load modeling, widespread faculty advising, and early warning or alert systems of individual student progress. Direct student access to their degree audit or advising records is problematic, but manageable (c.f. Lonabocker, 1989). The task of creating a structured set of program requirements and building algorithms to evaluate these requirements is, likewise, being refined and standardized (c.f. Darling, 1987). The individual student's expectations will soon change to include degree audit and automated advising support as a requisite part of an institution's basic resources and services.

References

Darling, A. L., 1987. A Generalized Table-Driven Degree Audit Algorithm, College and University, Vol. 62 - 3, pp. 265 - 270.

Lonabocker, L. 1989. U-VIEW: A Student Public Access Record Keeping System, College and University, Vol. 64 - 4, pp. 349 - 355.

Petersen, E. D., Kramer G. L., 1984. Proceedings of the Computer Assisted Advising Conference, Brigham Young University, Provo, Utah.

Southard J. 1989. Degree Audit Reporting System (Users Handbook), Miami, Ohio.

Spencer R.W, Petersen, E.D., 1983. Designing and Implementing a Computer-Assisted Academic Advisement Program, Jour. of College Student Personnel, Vol 24, pp. 513-518.

Systems and Computer Technology Corp, 1990. Information System Feasibility Study for Cuyahoga Community College (internal document).

Winston, R. B., Endler S. (eds.), Academic Advisement and Student Development: Helping Students Attain Educational and Personal Goals, 1984, San Francisco, Jossey-Bass.

CUSTOMIZED TOOLS

CAAD System (Computer Aided Application Development)

Wayne Ostendorf
Frank Maly
Iowa State University
Ames, Iowa

Customized tools is a so· ware system (CAAD) that Iowa State University uses to aid the programmıng staff in the development of application systems. The CAAD System is built around several ADP Center databases at ISU. These databases provide input for the CAAD generator programs that produce the source statements for application programs. This process eliminates much of the tedious coding or code copying, speeds up the development process, reduces costs and has positive effects on application development staff.

CUSTOMIZED TOOLS

CAAD System (Computer Aided Application Development)

Overview

The CAAD System is a set of productivity enhancement tools used by our application development and maintenance staff. The system consists of many separate units that work together effectively to help applications programmers get their work done. Each unit in the system is designed to improve the programmer's productivity by eliminating large portions of tedious program coding, producing better documentation, and facilitating the understanding and maintenance of the code.

Development

The model used in developing these processes has its basis in long term procedures and programming standards used in the ADP Center. The importance of everyone on the staff using the same organization and style in their processes is a principal concept of the tools.

While developing approaches to using the computer as a tool to aid programmers, it became apparent that much of the information needed to accomplish this was already available in other in-house developed systems. This project was also highly influenced by a team of students doing research on ways to improve analyst to programmer communications for coding programs. In addition to the objective of generating code with computer systems, we felt that improved documentation could also have a major impact on programmer/analyst productivity. Since documentation has a history of generally not getting done, the development of the tools was highly oriented toward having the documentation be part of the application development process. The system is thus designed such that the basic information about an application will document much of the system prior to development and will be used as input to the tools that generate the program source code.

During the early stages of development, we continually studied software tools oriented toward aiding system development that were from other sources. The tools available at that time appeared primitive for our use, required a lot of machine resources, were high in cost, and did not follow the ADP Center's internally developed standards.

Using the ideas from our previous experiences and from observation of the vendor tools available, we began to develop tools that could reduce many of the tedious tasks involved in application development. As we worked with the tools, we continually looked for input that could make the generated code more complete. We found that many datasets in the Center contained data that could be helpful in making generated code more complete.

The System

The System consists of seven major datasets and approximately 15 programs. Appendix A is a graphic of the general system flow. The main dataset, the File Master, in the system contains the primary information concerning files used by the tools. This dataset has a 7-character file ID key and contains the record size, record key information, file organization, and record description name. Other datasets used by the tools include a User Master File, System Documentation Master File, Program Title Master File, Analyst and Programmer Name Master File, Map Parameter File, and Data Dictionary File.

The input to the generator contains the ID for the program, system, file, user, and map. The ID for the analyst is available on the System Master. The ID for the author (programmer) is available on the program title master.

The generator program, a COBOL program, creates COBOL source statements that follow the standards developed by the ISU ADP Center programming staff. The code is very complete for the definition of files and the accessing of files. After the generating process, the programmer adds the source statements necessary to do the desired logical functions.

Because ideas and input were gathered from the entire professional staff, the acceptance level in using the tools is very high.

The System has been upgraded to support DB2 programming and now creates the desired source code instructions for DB2. This serves as a great aid in converting applications to DB2 because the application development tools take care of most of the basic routines. This allows programmers to write new programs much faster than when they have to learn all of the DB2 programming rules and use specifications.

Benefits and Costs

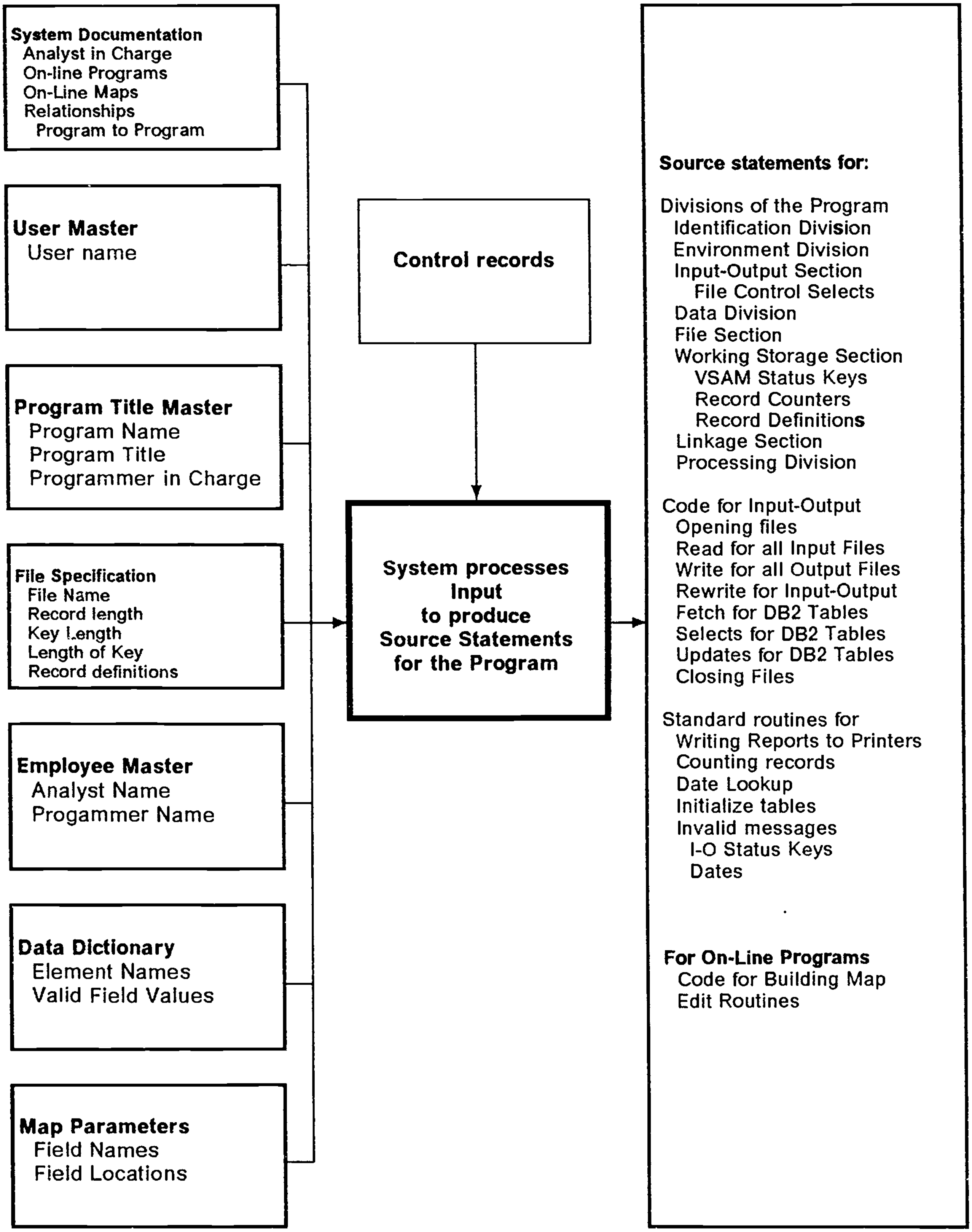
This project has proven to be very rewarding because it saves a lot of a programmer time and is heartily accepted by all programmers. Since programmers can improve their application turn around time, the users also become more satisfied. Another advantage is that the code meets standards for the ISU ADP Center. These standards make the code easy to follow because every COBOL program has the same organization, documentation content, and style.

The costs of the tools have been much lower than the costs of purchasing such software. The initial programming was done with part-time student employees; therefore, the cost has been kept lower than with full-time staff. These tools also give us a good beginning for any move to current or future CASE tools that may improve upon the CAAD System.

Conclusion

Most Data Centers have datasets similar to ISU's available to support the types of tools we have developed and use. We feel that much of the success we have in satisfying user needs is due to the support tools we have implemented. We are interested in sharing with others concerning their activities in this area and their future plans. Many of the available new CASE tools are large and offer good possibilities. Their present costs, however, may be high for many small-to-medium Data Centers; therefore, we may need to continue finding our own ways to help our staff members improve their productivity. Using automation to help us do our work is increasingly important to the future of administrative information systems technology.

COBOL Generator Flowchart



System Documentation
Analyst in Charge
On-line Programs
On-Line Maps
Relationships
Program to Program

User Master
User name

Program Title Master
Program Name
Program Title
Programmer in Charge

File Specification
File Name
Record length
Key Length
Length of Key
Record definitions

Employee Master
Analyst Name
Progammer Name

Data Dictionary
Element Names
Valid Field Values

Map Parameters
Field Names
Field Locations

Control records

System processes Input to produce Source Statements for the Program

Source statements for:

Divisions of the Program
Identification Division
Environment Division
Input-Output Section
File Control Selects
Data Division
File Section
Working Storage Section
VSAM Status Keys
Record Counters
Record Definitions
Linkage Section
Processing Division

Code for Input-Output
Opening files
Read for all Input Files
Write for all Output Files
Rewrite for Input-Output
Fetch for DB2 Tables
Selects for DB2 Tables
Updates for DB2 Tables
Closing Files

Standard routines for
Writing Reports to Printers
Counting records
Date Lookup
Initialize tables
Invalid messages
I-O Status Keys
Dates

For On-Line Programs
Code for Building Map
Edit Routines

A COMMON INTERFACE TO MULTIPLE APPLICATIONS

Candace Wilmot
Catherine Salika

University of Illinois
Urbana
Illinois

The University of Illinois runs a library circulation system and an on-line catalog which are two separate and unlike applications. We are about to install a bibliographic search engine, BRS, which has yet another command language. In addition, the libraries in Illinois for which we provide library services wish to access many commercial services. We need to provide a common interface in order to make access to all of these applications have the same "look-and-feel" and to provide assistance to naive users, similar to that which would be provided by a proficient reference librarian. Several different user groups have written various interfaces, both PC and minicomputer based. We intend to build on this work; provide the same functions from the mainframe for non-PC access; and integrate access to new services as required. We have developed a strategy to distribute the interface to various platforms, including MVS, DOS, OS/2, and AIX.

Background

2500 libraries in Illinois participate in the Illinois Library and Information Network (ILLINET) through 18 regional library systems, administered by the Illinois State Library. The computer system which supports these libraries is called ILLINET Online. It provides a union catalog for the 800 ILLINET/OCLC libraries in the state, circulation facilities for 38 academic libraries, and interlibrary access for all libraries in the state. ILLINET Online is directed by the Illinois Library Computer Systems Organization (ILCSO), which has 3 levels of membership:

A. Direct participants: institutions that maintain a current circulation database and agree to lend their materials to other ILCSO members. Patrons of these institutions may borrow materials from other ILCSO members.

B. Reciprocal participants: institutions that use an automated circulation system other than ILLINET Online and develop direct links between their system and ILLINET Online, allowing reciprocal borrowing.

C. Indirect participants: all other ILLINET member libraries. Indirect participants may borrow materials for their patrons from direct particpants.

The computer facilities and telecommunications network which support ILLINET Online are operated by the University of Illinois, through the University Office of Administrative Information Systems and Services (AISS). These consist of an IBM 3090-200J with about 1400 hard-wired terminals and 100 dial connections.

The funding for ILLINET Online comes from three sources: the Illinois Board of Higher Education (72%); the Illinois State Library (10%), and the ILCSO member libraries (18%). The annual budget for FY 1991 is $4,500,000.

Components

ILLINET Online is made up of several components. The Library Computer System (LCS) is the circulation component. LCS was originally developed by IBM for the Ohio State University Libraries and was the first component of

ILLINET Online, beginning state-wide operation in July 1980. LCS contains abbreviated bibliographic records which may be retrieved by call number, author and title. It supports a full range of circulation transactions: charge, discharge, save, and renew. It is used by direct participants to circulate material to their own patrons as well as to off-campus borrowers. The LCS database now contains records representing 10.5 million titles and 18.3 million volumes.

A second component of ILLINET Online is the Full Bibliographic Record (FBR) system. This is the online catalog, and is based on software developed by the Western Library Network (WLN) running under IBM's Customer Information Control System (CICS). FBR serves as a shared union catalog for the 800 libraries in Illinois that participate in the ILLINET/OCLC project. In addition to serving as a statewide union catalog, FBR doubles as a local public access catalog for the 38 direct participants. FBR records are complete cataloging records and may be retrieved by subject, title keyword, author, and other types of bibliographic information. The FBR database now contains 4.7 million bibliographic records and 7 million author/subject records.

Two other services are available. The Illinois Bibliographic Information System (IBIS) is based on a data base searching engine written by BRS Software Products. It can access a number of different commercially produced data bases. The two campuses of the University of Illinois are providing Current Contents, Medline, and eight Wilson data bases to their patrons. Plans are under way to extend access to some of these data bases to the rest of ILLINET Online in the near future. IBIS can search any text data base; most of the commercially available data bases are citations to journal articles, which are in great demand by the libraries.

The Colorado Alliance of Research Libraries (CARL) provides another search engine which runs against bibliographic data, text files, and other kinds of data. ILCSO has contracted with CARL for a service called UnCover, which provides an index to journal articles. CARL has included the ILLINET Online holdings information in their data base, so that when an article is located, Uncover will display libraries in Illinois which have that journal.

Network

The ILLINET Online institutions are served by a state-wide telecommunications network, using IBM's Systems Network Architechture (SNA). The network connects about 1400 terminals throughout the state with the computer facilities at the Chicago campus of the University of Illinois. Terminals are primarily ASCII terminals or PCs emulating ASCII terminals. Terminals at a given institution are multiplexed onto a single phone line by the IBM 3708 Network Conversion Unit. The terminals in the southern half of the state are brought back to a communications center in Urbana and then concentrated onto a high-speed link to Chicago. Terminals in the northern half of the state are connected directly to the Chicago location.

Differences in Applications

The ILLINET Online applications vary widely both in structure and in use. FBR has complete bibliographic data for items cataloged since 1974, but very little holdings information. LCS has complete holdings information for all items, down to the piece level, but has abbreviated bibliographic infomation. Neither FBR nor LCS has information about articles *within* the serials listed in the data bases. This type of article information is provided by IBIS and CARL, but IBIS has no holding information to show where the article might be found. CARL has holdings information for some, but not all, ILCSO libraries, but does not have call numbers.

The command languages also vary. FBR and LCS are command driven, although the command languages are quite different. IBIS and CARL are both menu driven, but there is little similarity in their appearances.

FBR, LCS, and IBIS are all IBM mainframe applications, while CARL runs on a Tandem mainframe. FBR and IBIS run under IBM's CICS telecommunications monitor; LCS is a stand alone application using TCAM and VTAM for terminal access. FBR is written in PL/I and assembler; LCS and IBIS are written in assembler. FBR, LCS, and IBIS are all run at the University of Illinois; CARL is located in Denver, Colorado, and a VTAM network connection is provided from it to the Illinois network.

Ad hoc solutions

With such diversity, it was difficult for the average user of library services to know which service he wanted, how to access it, and what to do with the information after he got it. To get around these problems, ad hoc solutions began springing up in the user community. A U. of I. linguistics professor produced a PC-based front end program which attempted to integrate access to LCS and FBR. It also provided assistance in formulating searches and navigating through the command language and it translated some of the coded information in the displays to a more readable form. It did this by emulating a standard library terminal and issuing existing LCS and FBR commands, as needed, and then combining the output into a more homogeneous display. This front end was widely used on the Urbana campus and was later distributed to some other ILCSO libraries.

When CARL and IBIS became available, a new version of this PC program was produced, taking advantage of software and hardware advancements to provide a library workstation which accessed the four applications mentioned above, plus other dialup services and some local applications.

The Computing Services Office (CSO), the academic computing center at the Urbana campus of the University of Illinois, provided a terminal server to allow access to ILLINET Online from terminals connected to the academic campus network. Again, it attempted to aid the user in constructing LCS and FBR commands, although the appearance was different from the PC program used in the libraries.

Although these user interfaces were developed independently of the mainframe applications, it was soon apparent that they were not truly independent. Since the interfaces relied on using existing commands, and on extracting data from fixed locations on screens, any change in the mainframe application had the potential to disrupt the operation of the interface programs. In fact, several mainframe changes had to be backed out when it was discovered that the PC interface would no longer work with those changes. The developers of these ad hoc solutions did not anticipate our widening range of services, nor were they able to provide adequate support for their products. It was clear that some sort of central coordination was required. In addition, there were many terminals in the network which could not run the PC interface and were therefore unable to get the benefits of a user-friendly interface.

Design of new interface

For these reasons, AISS decided, in October of 1989, to begin the design of a new, mainframe based, interface which could be used by all library patrons, regardless of whether they had a PC or not. This new interface, called MILO (Mainframe Interface to Libraries Online) would be written in a fourth generation language; it would take advantage of menus, help facilities, and pop-up windows as appropriate. Initially, it would provide an interface to LCS and FBR; in the future, it would be extended to IBIS, CARL, and other network services. MILO was designed to be used on the IBM 3270 display terminal, or equivalent, and its appearance was specified by librarians, with aid from AISS. To accomplish this, the ILCSO Operations Committee created an Interface Subcommittee, whose charge was to develop a prototype for MILO. The prototype, developed using DEMO II, prescribed the screen layouts, the sequence of menus and the format in which the data was to be displayed. In particular, the Interface Subcommittee specified the points at which the help function could be invoked and the wording of the help screens that would occur, based on their experience with common user problems.

This prototype defined a number of facilities which were extensions of existing operations. For instance, FBR could list books by Charles Dickens; MILO could give you the books by Charles Dickens published between 1850 and 1855. FBR could list all the books on aardvarks; MILO could select only those books on aardvarks written in German. MILO could also display information based on the form of the item. Where FBR could list all copies of Beethoven's Fifth Symphony, MILO could list only the scores, or only the sound recordings, as desired. The addition of filtering and sorting to MILO's processing allowed access to the data in ways which were not possible without MILO. The prototype also specified a seamless paging back and forth in lists, without the end user being aware of the specific commands being issued behind the scenes to produce the display.

While this prototyping effort was in progress, AISS, with the assistance of IBM, proceeded with the detailed design of how MILO would function. As the old applications all performed their respective tasks well and the effort involved to recode them was prohibitive, we decided that MILO would use existing commands to communicate with all applications. As the design proceeded, it became apparent that a few additional data formats were needed from FBR and therefore code was added to FBR to provide them. These data formats supplied blocks of data for MILO to process, rather than screen images to be displayed. Since this interaction was in the nature of a peer-to-peer relationship, the design called for something other than terminal emulation by MILO. IBM's Advanced Program-to-Program Communications (APPC) was chosen for communication between the two CICS regions (MILO and FBR) and between MILO and LCS. The modifications to FBR and LCS to add APPC support were relatively minor, localized changes.

Several benefits resulted from this design. Coding of the various parts of the system could proceed in parallel, thereby shortening the development cycle considerably. In the future we would be free to modify the applications as necessary, because MILO could shield the users from these changes. The change from the ASCII terminal standard to the 3270 made available other network services that were written for 3270 terminals.

The fourth generation language chosen for writing MILO was NATURAL from Software AG. As FBR uses Software AG's ADABAS for its database management system, NATURAL was already available as a tool. We felt that using NATURAL would provide substantial productivity enhancements in the development of MILO over using PL/I or assembler language.

Although one major benefit of this approach was the minimal changes necessary to the older applications, we actually have a long-range plan for major changes to LCS. In the future, we intend to move the bibliographic information out of LCS into FBR. Consequently, all searching for items will be done using FBR search commands rather than LCS search commands. By having MILO in place before doing this restructuring, it can take place entirely behind the scenes. The average user will be totally unaware that the familiar search results screens are being created with a different set of search commands.

Implementation

We divided the implementation effort into several areas. The coding on MILO began first, programming the user screens as defined by the prototype. Using a stub to retrieve a few sample records of each format, work was able to proceed although MILO could not yet communicate with any of the other applications. At the same time, we contracted with a consulting firm to provide the basic APPC routines to communicate from one CICS to another CICS and also to a stand-alone VTAM application. This APPC code was then incorporated into LCS using a simple driver program from CICS. The remaining link was then put in place so that the MILO code could call, via APPC, on LCS for its test data.

At the present time, the MILO coding is about 2/3 complete, the LCS modifications are complete, and the FBR modifications are in progress. We have made a test version of MILO available to the Interface subcommittee so that we can get feedback as the development proceeds.

While the mainframe interface was being developed, a project was underway to prepare a PC interface program which would use MILO to access LCS/FBR, and would also provide sophisticated searching assistance in accessing IBIS and CARL. This PC program also provided a framework for local library information services and local data bases of interest to a specific clientele. It was developed at the Urbana library, and will be packaged and distributed by AISS to the ILCSO libraries and to individual faculty and students.

Distributed Processing

The MILO design, although running on a single mainframe in the first implementation, is capable of being distributed tc any platform which supports APPC. This would allow MILO to be ported to other mainframes (or to PCs) at some time in the future. We have the option of running multiple copies of MILO, on different CPUs, at different locations, if desired for better performance. We could also run multiple copies of any of the search engines, if that also becomes desirable as the system grows.

Much work has been done on the PC interface used by the Urbana library, particularly in the area of local services and a user-assisted IBIS dialogue. To bridge the gap until MILO is fully functional, this PC program will be distributed to the rest of the ILCSO libraries. As MILO is phased in, the ILLINET Online assist features of the PC program can be reduced and the local functions enhanced. In a later phase, this PC program can be changed from a terminal emulator to a peer communicating directly with the other applications: a PC version of MILO. This version would be able to take advantage of advances in PC presentation software to produce results that would be unobtainable from mainframe-based MILO alone.

Conclusion

We have concluded that you can blend several older systems into a seamless, apparently new, application with much less effort than it would take to replace the older applications. This approach makes sense if the underlying applications are sound. In our case, LCS and FBR were very flexible and efficient search engines. What made them appear dated was their old fashioned commands and displays. We are realizing the productivity improvements that go with a fourth generation language, without having to rewrite the entire applications. We think this is a good start to a major refurbishing of one of the applications, as MILO will preserve the appearance of that application while the structure changes radically.

We felt that it was important to get control over the other interfaces, either by providing an improved version or by providing a new facility that made them obsolete. This gave us a lot more flexibility to make changes in the system without the fear that some unknown interface would stop working. The easiest way to provide this new interface was on a mainframe, but we wanted to be able to take advantage of distributed processing at some later date. We intentionally designed MILO to that parts of it could be distributed to workstations in the future.

Finally, we were fortunate to have a group of resourceful librarians to develop and test partial solutions to the interface problem, so that they were able to give us some good advice when the time came to develop MILO. We think the result will be a state of the art system for our user community: the libraries of the State of Illinois.