This paper was presented at CUMREC '97, The College and University Information Services Conference. It is the intellectual property of the author(s). Permission to print out copies of this paper is granted provided that the copies are not made or distributed for commercial advantage and that the title and authors of the paper appear on the copies. To copy or disseminate otherwise, or to republish in any form, print or electronic, requires written permission from the authors.
Interfacing to the Internet: Re-thinking Administrative Systems

Abstract

Developing student information systems on the Internet opens opportunities to re-think what we are trying to deliver with these systems and to whom. In the past, administrative systems have been developed to manage large amounts of complex information. Because the data is so complex, access has been tightly controlled through a small number of interfaces. For example, student information systems typically have had several character based terminal interfaces for staff and telephone or kiosk interfaces for students.

This is in direct contrast to what people have been experiencing through the interface of most personal computers. The standard desktop abounds with objects such as calendars, file folders, spreadsheets, schedulers, mailboxes and garbage can’s. The newest addition to this rather uncontrolled environment is browser access to the Internet. Often, hidden away on this desktop is several icons allow staff access to the old character based administrative systems.

Internet systems offer a way to organize this chaotic environment in a new way. The underlying philosophy of Internet systems is wide availability and simple interfaces. If it were possible to use the familiar interfaces seen on desktop objects, extend them to administrative systems problems and finally, distribute access to them over the Internet, a substantially different way of delivering administrative systems would be at hand. The development of new tools and techniques for Internet systems suggests this is now possible.

This paper is a study of what some of the alternatives and implications might be if this melding of the personal computer desktop continues with both the Internet and our administrative systems. Using examples developed for the Internet, this paper will attempt to show what the interfaces of student systems in the future might look like. Issues discussed will include what the implications might be for developers, users, administrators and universities.
Access and Understanding

University administrative staff, students and faculty have always had difficulties getting access to enough institutional information to make informed decisions. This information consists of rules, regulations and general information written in a variety of university publications such as calendars, admission guides and other similar documents. With the introduction of computers, information is further divided between paper publications and multiple computer formats. Often, the final link in access to information is a knowledgeable and experienced administrator.

Access to information is one issue. Understanding complex information is another. Ask any student or faculty member if the rules and regulations governing their institution are easy to understand and in all likelihood the answer will be no. From fee assessment to registration, from admissions to financial aid, the rules in these and other areas are complex and difficult to administer. Consider the questions a student must answer when choosing a course for their degree program: Will a course help me complete my degree? Does it conflict with other courses I am taking? Is my average high enough? Can I drop another course to make it fit? Does it look interesting? Do I meet the requirements? Both, new and experienced staff and students have to use computer systems, visit administrative offices, search publications and then synthesize all this information to answer these basic questions.

The result of these problems of access and understanding is the existence of the bureaucracy seen at all universities across North America. Offices and departments for admissions, advising, awards, financial aid, room scheduling, course scheduling and registration, among others, contain specialists who manage the access to and understand the complexities of the information. Students, staff and faculty turn to these specialists to solve their individual problems. These offices by their very nature become bottlenecks because demand for their services are so high.

Evolving Administrative Systems

The introduction of computers over the last four decades has allowed administrative departments to tackle these problems in new ways. Over time new systems have enabled universities manage their information in even more complex ways while increasing access to that information. The changes started in the late 1960’s and early 1970’s when punched cards and transactions on magnetic tapes were used to update central databases. Then in the late 1970’s and early 1980’s character based terminals were used to communicate to central mainframe databases. Despite this computer penetration, large amounts of printed information continued to be generated: university calendars, course schedules, admission guides and the like.

In the late 1980’s and early 1990’s access to administrative information suddenly expanded. Personal computers were introduced. Telephone registration systems rapidly became the primary interface for students to register in courses. The first graphical user interface client server systems were released. Touch screen kiosks became common place on campuses. And, most dramatically, in a few short years Internet access to administrative systems has gone from a dream to a critical part of service delivery at many institutions.

In a few areas the problems of access and understanding have been solved by technology. With telephone registration, touch screen kiosks and the Internet, students are able to manage many complex administrative transactions on their own. These developments have helped students but have resulted in a requirement that these general users be familiar with several different representations of the underlying information -some is accessed through the telephone, other information is presented on kiosks and now some information is manipulated through the Internet. These developments have also created a management problem for administrative staff who still use character based interfaces into administrative
information yet find themselves managing this eclectic set of telephone, kiosk and Internet interfaces for students.

The Internet may provide an opportunity to unify all these interfaces for both students and staff into one common interface. With global access fast becoming a reality, it is the network that will let anyone and everyone be connected. The key issue has made this unworkable until now has been the inability to create sophisticated user interfaces on the Internet. Now with programming languages such as Java and with extensions to standard development languages, it is very plausible to consider developing rich system interfaces on the Internet.¹

**Taking Advantage of Computer Literacy**

Despite re-engineering efforts, very few institutions have been able to reduce the complexity of the rules and regulations that govern them. Most often increased complexity has been handled by building bigger and better processes or computer systems. Little thought has been put into re-thinking how and by whom these processes are done. Much of this falls into the business process re-engineering category. However, regardless of this, at the core of most administrative business processes is a large set of complex data that is difficult to maintain and manage by its very nature. Our computer interfaces into this information reflect this complexity - they are typically difficult to use and understand.

Ironically, the desktop computer is much like these administrative systems. Underlying a typical 1990’s computer is an incredibly complex environment that requires sophisticated software to manage it effectively. To deal with this complexity, interfaces to desktop computers have evolved rapidly since their introduction in the mid-1980’s. They have gone from command line driven environments like Microsoft DOS to the point and click graphical user interfaces of the Apple Macintosh and Microsoft Windows. In contrast to administrative systems, these desktop environments are now easy enough to understand that we expect grade school students to be adept at their use.

If one assumes that that these desktop interface changes have enabled large numbers of general users to easily take advantage of complex desktop computer environments, could similar interface changes be used for university administrative system interfaces to reduce their complexity? This may not be unrealistic given that many people are starting to expect their interactions with all computer systems to work with the same common set of objects and behaviors as a result of this evolution of the desktop. Some of the common objects used regularly and that many people are becoming familiar with include:

- Personal time schedulers
- Electronic mail packages
- File folders
- Calculators
- Spreadsheets
- Search engines

Sitting down at another person’s desktop, one almost expects to see one or several of these icons. Understanding an object on a desktop means, in part, understanding what it represents. However, it also means knowing how and when to interact with it. Some of these objects embody metaphors that ante-date the computer (such as schedulers, folders and mail). Others such as calculators and spreadsheets owe their very existence to the emergence of personal computing.² This understanding of the desktop has become what many call computer literacy. In many organizations this basic literacy is a requirement for employment.

---

¹ See appendix A for a short discussion of the types of tools available
² Ironically, the desktop itself is a metaphor that antedates the computer by hundreds of years!
Desktops on the Internet

If one were to take these basic, almost trivial objects and apply them to more complex administrative problems, one could leverage off this common computer literacy to deliver easy to understand systems. What is necessary is to imagine scenarios such as the following:

- Using tools such as personal desktop schedulers and extending them for course and room scheduling
- Considering the use of spreadsheets as truly integrated student system grade books
- Using metaphor of desktop file folders to contain student information
- Expanding Internet search engines to query administrative systems as generic reporting tools.

If examples like this were possible, mastering a particular business process, in the sense of having to learn complex sequences of key strokes or having to learn the operations more closely tied with the requirements of technology than of the business, might no longer be an issue. Basic computer literacy would be the issue.

Then, if one were to add in the ability to distribute these new interfaces over the Internet to anywhere in the world, it could become an attractive option to consider using one interface for both basic and sophisticated system users. Like our current administrative systems, sophisticated users would have access to drill down through the objects to the more complex aspects of the data but the same familiar interface would be the starting point for all users.

We believe these ideas are not only possible, we think they may be the way administrative systems are delivered in the future. They may sound far fetched but consider some examples from the recent past. In the 1970’s highly trained individuals were required to do word processing for large offices. Now, with graphically rich software widely available, almost anyone can do basic word processing tasks. In the 1980’s trained individuals were required to do data entry on administrative information systems. Now, with the point and click interface of the Internet, we often see students doing their own data entry. Commonly used desktop objects that are repeated in different contexts combined with regular usage have made what were previously considered complex tasks easily achievable by untrained individuals.

Scheduling

Description

Take for example course and room scheduling. Calendars, schedules and timetables are the simple instruments we use to plan future events. They are represented in a number of different ways on the personal computer desktop. However, existing desktop schedulers suffer from a number of limitations that would impact their use if they were to be used in course and room scheduling. Their primary limitation is that they are designed to schedule personal appointments. These tend to be unique events, rather than events that may follow complex scheduling patterns. Further, the events being scheduled are fairly trivial from a logistical point of view - for example, a meeting of certain people in a given room at a specified time.

There are none of the complex dependencies normally associated with developing course and room schedules. In order to overcome these limitations, a desktop scheduler extended to work in this environment would have to be able to:

- attach a schedule to any object: a staff member, a student, a room, a course section, an exam or even a piece of equipment.

---

3 A common task many universities now have applicants and students do over the Internet is apply for admission and change addresses.
• have a simple and intuitive method for developing recurring and potentially unusual schedules; and,
• be able to overlay the schedules of several different objects. This is necessary in order to find out which rooms out of a specified set may be available to meet the requirements of a given schedule.

Such a course and room scheduler modeled after standard desktop schedulers might look as in Diagram 1.

Diagram 1
The Scheduler

In order to find out when a particular event can be scheduled it is necessary to overlay the schedules of the things/people that are required to participate in the event. This can be done by:

1. Opening all the schedules of those events/things
2. Highlighting the schedules one wishes to overlay.

The middle section of the schedule is, typically, a daily view of a particular set of objects (e.g. the schedule on Monday). However, it should be possible to “zoom out” of this particular view to see a weekly, monthly or yearly view of the same schedule(s).

Scheduling on the Internet

In the short term, using the same look and feel of desktop schedulers for room and course scheduling might enable administrative units to reduce expenditures on training. However, if that same scheduler were distributed over the Internet, not only would it enable administrative staff to do their business anywhere and anytime, but it might also facilitate usage of the schedule by large numbers of people who previously were not considered online users of a schedule.
This new schedule would have to be designed so that infrequent users could use it in similar ways as desktop schedulers are used - simply to determine when people, courses and rooms were booked. More sophisticated users would have broader access, as defined by security rules, to drill further into the schedule to access and update information appropriate to their level.

With a bit of imagination, it is possible to consider using an Internet scheduler to handle such scenarios as the following:

“Let us assume that everyone makes their schedule as public as possible. A student could open a professor’s schedule and directly book an appointment without conflicting with a course section he is teaching (because schedules are delivered through the Internet they are available at any place and any time). A professor’s schedule would always automatically reflect his/her teaching engagements because the schedule is directly connected to the course database. Alternatively, individuals wishing to book a room on an ad hoc basis could do so through the Internet using the scheduler by simply opening the room schedule and booking a time while paying for the booking.”

These examples are simple illustrations of what it means to have a widely available common and simple interface to a complex data such as course and room schedules. Rather than character based data entry screens isolated to highly trained individuals, the goal would be to open scheduling up to as many people as possible.

The Scheduler Extended - Timetables

Description

The desktop scheduler example extends reasonably well to encompass a standard problem faced by students at every university: developing a personal course timetable. Because the timetable is a powerful visual organizing symbol for students, one can imagine using a scheduling system to solve four problems:

1. Assisting a student in deciding upon interesting and appropriate courses
2. Discovering which courses meet program requirements.
3. Fitting the courses into a timetable around their personal and work schedule
4. Registering in the courses

A timetable defined in this manner becomes a powerful planning tool for students.

Traditionally the timetable is used to record a schedule once all the relevant booking have been made. Diagram 2 shows how the “interactive timetable” might actually become a vehicle in the process of making those bookings.
The student can initialize the timetable by indicating preferred times. This is done by blocking out unavailable times and optionally those times a student might not want to schedule classes. Courses or sections are added simply by highlighting them. By double clicking into particular sections, course descriptions and requirements are readily available for viewing. Other areas of the course schedule can be searched by using a simple search engine (see below). Clicking on the register button might be the final step in the process of the timetabling exercise.

Timetabling on the Internet

Introducing telephone registration at many institutions moved the registration period from a few weeks prior to the start of a session or term to several months before the start of classes. This required course schedules and programs to be ready for students several months earlier than in the past. The benefit was that administrators were able to get a better understanding of what the demand was going to be like often several weeks before classes began.

Introducing timetabling of this nature over the Internet opens new possibilities of planning for both students and administrators. Students would be able to gather all the information necessary to choose a course section from a single place: description, time and place, course and degree requirements, implications for their particular program of study and register at the same time. Administrators would not only have the planning tools they have with telephone registration they would be able to broadly distribute important information to students via a richer interface.

One might even imagine that students submit proposed timetables at the time of admissions or many months in advance of normal registration periods. Such a change might enable administrators to change planning processes from the current supply driven model (create a large number of sections, if there is not enough, create more) to a more demand driven model (create the number sections students have proposed registering in).
File Folders

Description

Personal records (birth certificates, bank statements, insurance etc.) are typically kept in file folders. The folder has become the universal symbol for the desktop computer’s filing system. In the context of student systems, personal folders can be built for students that include:

- biographical information such as name and address
- financial statements such as fees, fines and scholarships
- academic record such as transcripts and degrees

Each of these objects often actually corresponds to paper documents that students either send to the university or receive from the university. The electronic equivalent to a paper folder seems like an obvious visual symbol to organize this information.

When the folder is opened one might expect to find the information separated by tabs. Such a folder becomes an important tool that people at all levels across campus suddenly are interested in for day to day work. Diagram 2 shows a sample representation of a student folder.

### Diagram 3
**The Student Folder**

Once one starts thinking more about the folder concept it seems like many documents and document-like objects can be stored in such a folder. At the University of British Columbia the Undergraduate Admission System folder, a prototype of the one discussed here, also contains electronic data interchange transcripts and electronically received TOEFL scores. There is no reason why the folder should not also include imaged documents and e-mail. Indeed, it can be seen as a comprehensive record of a student’s interactions with the University.

Folders on the Internet

Making personal folders available on the Internet reveals another aspect of distributing these easy to use objects on the Internet: controlling access. By deciding to attend an institutions, students enter into an
agreement that allows institutional staff to access their personal information. However, beyond this institutional right, most institutions have strict policy’s outlining who else has access to this information.

An Internet based personal folder would have to be designed such that not only was the institutional right to access the information maintained, but would also have to enable students to control access to their information. This access control would have to be flexible enough to accommodate people and groups of people within the institution as well as people and organizations outside of the institution. The design of such a folder would have to include the ability for students to update basic information such as their addresses while at the same time enabling them to look at all the other non-alterable information maintained by the university such as transcripts and correspondence to and from the students.

Building such an Internet folder might not only enable better management of information within an institution, it might also facilitate communication between organizations. For example, rather than requesting a transcript or requiring confirmation of registration, students themselves might be able to give external organizations permission to view some or all of their information. This would give these other organizations the ability to determine the answers to their questions without requiring effort by departments within the university.

Search Engines

Description

Anyone who has hit the Search button on a Web browser has been introduced to the “Search Engine”. These “engines” are fast becoming popular not only on the Internet but also within organizations as mechanisms for finding information. The Web can be seen as a large and distributed database. The search engines all provide different ways of indexing and searching that material. The important point is that 15 years ago searching a large electronic database was a highly specialized skill limited to a small number of experts. Now the interface has been simplified to such a point that it is reasonable to expect patrons of a public library to be able to conduct their own searches.

There is no reason why corporate or institutional databases should not be made accessible through search engines. Diagram 4 shows one instance of the Search Engine object. This particular instance is used to search for educational institutions.

Diagram 4
Institution Search

<table>
<thead>
<tr>
<th>Search</th>
<th>Institution Name</th>
<th>Institution Type</th>
<th>Province</th>
<th>Country</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clear</td>
<td>Carson Graham High</td>
<td>Secondary School</td>
<td>BC</td>
<td>Canada</td>
</tr>
<tr>
<td></td>
<td>Cariboo</td>
<td>Secondary School</td>
<td>BC</td>
<td>Canada</td>
</tr>
<tr>
<td></td>
<td>Charlie Elementary</td>
<td>Secondary School</td>
<td>BC</td>
<td>Canada</td>
</tr>
<tr>
<td></td>
<td>Central Church</td>
<td>Secondary School</td>
<td>BC</td>
<td>Canada</td>
</tr>
<tr>
<td></td>
<td>Cent Hill Secondary</td>
<td>Secondary School</td>
<td>BC</td>
<td>Canada</td>
</tr>
<tr>
<td></td>
<td>Crescent</td>
<td>Secondary School</td>
<td>BC</td>
<td>Canada</td>
</tr>
</tbody>
</table>
The search engine object has these characteristics:

- it is, by definition, a keyword search
- boolean logic is implied in that all the conditions are linked by an AND.
- it is independent of the characteristic of the object we are looking for. For instance, once we have found the student, we still have to decide whether we are looking for the student’s address or transcript, fees or other attributes of a student.

A common feature of search engines on the Internet is the so-called “advanced search”. This implies the ability to supply more complex boolean expressions and thus by-pass the pre-formatted boolean logic of the simple search. Again, there is no reason why this facility should not be supplied for large institutional databases. Diagram 5 illustrates and advanced search engine.

![Diagram 5: Advanced Search Engine](image)

### Search Engines on the Internet

Making search engines of this nature available on the Internet takes advantage of a metaphor that people everywhere are becoming familiar with. Applying it to the domain of student systems, especially advanced search engines, enables faculty and staff to search on categories of students they previously had to rely on central offices such as the Registrar’s Office to generate. These search engines, in a sense, become fully distributed reporting tools.

### Other desktop tools: Calculators, Spreadsheets and E-mail

So far we have only examined a very small sample of desktop tools. There are, of course, many others that could be used to simplify access to our administrative systems. A few examples might include:

#### Smart Calculators.

A smart calculator would look just like the normal desktop calculator but, because it would have access to the institutional rules database it would be able to do things like calculate fees and grade point averages. Such a calculator might not only be available for internal use but also for students confirming their fees or admission applicants checking their admissability.
Spreadsheets

Spreadsheets delivered over the Internet could become the vehicle for updating and maintaining grades on the central administrative systems. This would not be in the sense of uploading and downloading data into spreadsheets but, rather, opening a spreadsheet directly into the administrative database. Spreadsheets are already heavily used at universities by faculty for grades collection; what has been missing is a direct connection into the student system.

E-mail

E-mail can be used for a variety of communication within and outside an institution. A common use many institutions are now extending it to is to exchange official correspondence with students rather than sending or receiving the paper equivalents.

However, e-mail can be used in other ways. An issue faced at many institutions is automating workflow. Commercial applications tend to use a mixture of electronic forms and electronic mail. Two problems are solved via this method: 1. Passing information from one person to an other in some order of preference, and; 2. Signing off on that information. Unfortunately, many of the solutions available today rely on proprietary forms. However, if the information were available everywhere in easy to use forms, that is on the Internet, the only thing that becomes necessary is notification that information has changed and signing off.

An example this approach would be course approval over the Internet. Given there is an Internet accessible course database, all that is necessary is that the various levels of organization review, revise and approve changes. No forms are necessary, only direct updates to the Internet database and e-mail notification of subsequent approvers. A working example of this approach was presented at CUMREC 96.4

Security

Many of these ideas are simply meant to be thought provoking. Others, are already implemented at some institutions. Some are starting to be available over the Internet. Creating systems on the Internet with desktop objects allows one to build extremely powerful systems. It also opens up security issues that can not be ignored.

It is not within the scope of this paper to argue the technical merits or demerits to distributing secure systems on the Internet. Suffice it to say it is the belief of the authors of this paper that the efforts of Netscape, Microsoft and other commercial vendors are going to make secure interaction on the Internet a reality in the near future. Whether it be electronic certificates of authentication (e.g. SSL) or electronic tickets issued by particular services (e.g. DCE), the solutions exist. It is simply a matter of selecting a solution that best meets an organization’s needs.

It is, however, appropriate to briefly discuss an approach to security or a set of security principles that would make these easy to use Internet based solutions work more effectively and differently solutions from the past. The historical approach to security for administrative systems security has been to tightly control access by closing off external access to these systems. Rather than carrying this philosophy forward to Internet based systems, it might be more appropriate to assume that all systems are inherently open and available. It becomes the information owners’ concern as to whether they would like that information to

---

be available. Two issues must be resolved here. First who are the owners of the information? Second, how to develop an interface that easily enables those owners to control access to their information.

The first problem, determining the owners of information, is likely the most difficult to resolve because most administrative systems have been based on the information owners turning control of their information over to some central authority to manage and maintain (e.g. the Registrar’s Office). However, this new Internet model would turn control over personal information back to students; control over room bookings to the building owners and course scheduling perhaps even as far down as the professors themselves. Obviously these changes would be difficult to manage but the implications should be considered in the design of any system.

The second problem, developing an easy to use interface for information owners to use, is also difficult but not impossible to solve. One suggestion would be to take the rapidly evolving security mechanisms that are being released for personal computer desktops and applying them to administrative data. Permitting people to access and update one’s information should be as easy as permitting someone to access a file on your personal computer. These mechanisms may be difficult to use today but because of the rapidly evolving desktop and Internet environment, it is not difficult to imagine user friendly permutations in the near future. What becomes more important in this environment is educating people about security concerns rather than explaining how to technically manage the problem.

Changing Roles

As noted throughout, many of these common desktop objects simply provide a powerful visual representations of underlying administrative data. What becomes more interesting with widely understood, freely available representations of this information, is that no longer is the maintenance of this information necessary by highly trained individuals. Rather, the true end users of information, whether it be students updating their registration information, Deans, Registrar’s, vice-presidents reporting on complex data or even professors scheduling courses, they would in this new environment have the ability to manipulate and control their own information.

If this were to happen on a wide scale at an institution, it might require a significant changes in the way administrative staff, information system professionals and clients or students work together. Some potential implications include:

**Central administrators.** Admission officers, course schedulers, front counter staff and other administrative staff might see their roles change from the funnel into which information flows and is dispersed, to dealers or arbiters of information and managers of the overall process of information gathering. They might ensure data integrity by reviewing end results rather than controlling the inputs. They might best act as a clearinghouse for complex problems that could not be handled by the routine system tools.

**University clients or customers.** Students, applicants, professors, deans, vice-presidents and others might become fully responsible for their own information rather than turning the responsibility over to others. This responsibility has two aspects to it: first, keeping the information up to date and accurate and second, as discussed previously, ensuring the appropriate security of their own information.

**Information system professionals.** Programmers and analysts might no longer build large systems for controlled groups of users. They might build small, discrete tools such as those described in this paper or perhaps data interfaces to be used by existing commercial tools for staff, students, faculty and the general public. Their expertise would begin to focus more on how easily are they able to integrate
institutional data into familiar desktop objects and tools.

**Conclusion**

Using the familiar interfaces seen on personal computer desktop, extending them to administrative systems problems and finally, distributing access to them over the Internet does not necessarily imply some of the ideas put forward in this paper. However, the implications of these developments do lead to some startling new ways of doing business at universities. Broadening access to levels achieved by the Internet and simplifying complex administrative environments to a level where true end users of information can actually use that information, may allow organizations to do business in ways never before imagined.
Appendix A - Developing Systems on the Internet

Developing common desktop objects for the computer systems has become almost trivial using tools like Microsoft’s Visual Basic, Borland’s Delphi or Powersoft’s Powerbuilder. These tools are enabling system developers to take desktop objects and put them into more complex applications. Until recently these products and many others like them have been focused on deploying the resulting applications on isolated desktops, not on the Internet. Now with the advent of Internet programming languages like Java many of these vendors are now re-working their tools to allow deployment over the Internet.⁵

At the same time, due to both hype and reality, the Java language has evolved from an interesting toy programming language for the Internet to a properly supported development language. Development tool releases from prominent vendors such as Microsoft, Oracle, Symantec and Borland are making complex systems development in Java possible. Lessons and techniques that have been learned over years by other development languages and vendors have been copied and augmented in these new Java tools at an incredibly fast pace.

What this means to administrative system developers is that systems development on the Internet has arrived. System features that at one time were relegated to specifically configured personal computers or that required special database connectivity can now be deployed over the Internet, anywhere in the world on any type of computer. It may be early to be making such claims but this appears to be the direction computing is taking us in the late 1990’s.

---

⁵ Java was created in 1995 by Sun Microsystems. It is a programming language used primarily for Internet applications.