The Development and Delivery of an Academic Departmental Management Information System through the use of World Wide Web Technologies

Ryan Cherland  
Principal Analyst  
University of Kansas  
Office of Institutional Research and Planning  
P.O. Box 505  
Lawrence, Kansas 66044-0505  
irrmc@ukans.edu

The University of Kansas, Lawrence, Kansas:  
“The University of Kansas is a major educational and research institution with more than 28,000 students, 1,900 faculty members and 7,000 staff and support personnel. In fall 1995, KU enrolled students from all 105 Kansas counties, all 50 states, the District of Columbia, 3 U.S. territories, and 109 nations. Two out of three KU students are Kansas residents. The main KU campus is on Mount Oread, a steep, tree-covered ridge in Lawrence, a city of about 68,000 in hilly northeastern Kansas. The university's other campuses are the Medical Center in Kansas City, the Edwards Campus in Overland Park, the Capitol Complex Center in Topeka, and a clinical branch of the School of Medicine in Wichita. KU is the largest of the six state-supported Kansas Board of Regents institutions. Within the regents system, KU grants a third of all bachelor's and master's degrees and 70 percent of doctoral and first professional (law, medicine, and pharmacy) degrees. Nearly half of the Lawrence campus budget and more than a third of the medical center budget come from state appropriations. The remaining sources of the university budget are gifts, grants, hospital revenues, and fees.” (extracted from http://www.ukans.edu/gateway/aboutKU.shtml, December 1996).

Abstract  
This paper describes how the World Wide Web is being used to provide management information to decision-makers of academic units. A system called DEMIS, Departmental Executive Management Information System, was developed to give its users historical trend data tables both as HTML pages and spreadsheets that could be maintained and updated via SAS® programing. As the system developed, additional functionality was added that allowed users to create their own queries to run against the analytical databases of the institutional research office. The various stages of development of the system as well as some of the security issues for this type of system are also discussed.
The Development and Delivery of an Academic Departmental Management Information System through the use of World Wide Web Technologies

Introduction
In these times of scarce resources and increasing changes within society and academe, administrators of academic departments have been asked to do more with fewer resources. Cutbacks in funding, increases in tuition and other fees, and changes in what learning and degrees will be valued in the future have lead to a more competitive market for students and created additional difficulties in managing academic departments. As George Keller (Keller, 1995, p.59) states, “Perhaps the biggest change for administrators will be from presiding over continuity and tradition to entrepreneurial management of the social, economic, and educational transformations of society and academe.” Academic deans and chairs are asking institutional research offices for increasingly more information on their units (Kinnick, 1994), where in the past, when resources were plenty they may not have known that an institutional research office existed nor cared.

Increased Need for Academic Planning and Management Information
The University of Kansas (KU) experienced a convergence of factors in the past few years that created a groundswell of data requests from academic units (college of liberal arts and sciences, school of architecture and urban design, school of business, school of education, school of engineering, school of fine arts, school of journalism, school of law, school of pharmacy, and school of social welfare). Through a combination of personnel changes and restructuring at all administrative levels of the university — at the chancellor level, the provost/vice chancellor level, and the dean level — the university has seen some new faces and some familiar ones in new positions. Many of these individuals needed information in a summary format for a quick understanding of the units for which they were now responsible. In addition, changes were occurring in how KU (and consequently the college and schools) were being financed. The state governing board, the Kansas Board of Regents, implemented a linear tuition system where students paid for every credit hour that they took; there were no longer any plateaus where students could take additional credit hours without having to pay additional tuition. KU also looked at how it financed the academic units and examined additional ways to make the units more entrepreneurial. For certain schools this meant that they could adjust their offerings at various locations and times-of-day to maximize the attraction to students and increase the enrollment in the schools’ credit hours. Any additional tuition income over a base level of credit hours would go into the schools’ coffers.

This created a need for information that was student credit hours based, not head count based, which was how much of the traditional student reporting systems were set up. Also, information was needed at least weekly. And requests for such information were coming at the institutional research office from many directions, not just the deans of the various schools, but also from the Provost area and the Budget Office. These factors created a synergy that pushed forward the advantages of using advances in information technology, and specifically Web based technologies, as the best method to provide the needed data. There are four reasons why it is advantageous to an institutional research office to provide information in this manner:

1) Having to deal with multiple requestors with different viewpoints of the information: The new system allowed each user group — provosts, directors, vice chancellors, and deans — to follow their own concerns and interests in the data without having to work through pages of computer printout. From the viewpoint of an office of institutional research, data could be provided so that: (1) once updated, information was instantly accessible to decision-makers; (2) it reduced the need to try to second-guess the needs of decision-makers; (3) it allowed decision-makers to instantly follow their hypotheses about changes in student course taking behavior without having to request a new report.
2) Avoids ‘Reinventing the Wheel’ in the Institutional Research Office: In an office with multiple analysts, all capable of writing programs to answer the types of questions that arise, the new system offers a single framework for users to query the information. This avoids the problem of analysts writing programs or making changes in programs to get a specific analysis that had not been anticipated. Although such requests often mean just changing the variable of analysis in a program, there are always some costs in time for the analyst — whether it is rewriting major sections of code, or simply setting up a new run, printing output, and writing an appropriate cover memo. Such activities act as interruptions and delays to other analytical undertakings.

3) Others with Informational Needs not Knowing what is Available: There are often misconceptions about the purpose of institutional research offices and who is an appropriate customer. Many deans and chairs often believe that such services are only available to the central administration of the university and not the academic units themselves. Or they are so consumed by situations in their areas that they have little time to frame their questions and put them in a memo or phone call. Some academic administrators are just not analytically inclined, but they still need to be able to find information that helps them manage their units. The new system allows these users to examine what other academic decision-makers viewed as important managerial variables and indices without a huge investment in time and resources.

4) Distribution and Retention of Reports: Academic units have been known to discard computer reports that are not from the current year. Yet, they often never think to put the information in a spreadsheet or data base in their office, so they end up with no historical data about their enrollments, teaching loads, or credit hour production. Luckily, institutional research offices seem to never throw things away, and can often provide the desired information when requested. However, the new system avoids these types of requests by providing decision-makers recent historical information to compare with current data.

By now most university staff have had experience ‘surfing’ the ‘net. As institutions put in place campus networks and as software browsers, operating systems, and hardware became more user friendly, ever more campus staff have ventured out onto the World Wide Web (WWW). Once the initial thrill of all the bells-and-whistles dies down and the need for the most recent U.S. weather satellite map or Dilbert cartoon doesn’t seem as important as it once did, the question becomes ‘Now what do I do with this?’ Campus offices that provide information should take advantage of this situation and make use of this new and evolving technology to help promote access do data via a campus intranet. An intranet differs from the Internet in one special way — access into it is limited. Tabke (1996) describes how this is done, “Although intranets are based on Internet technologies, these networks typically are privately managed with the intent of having industrial-strength service levels and security. Connections to the Internet are made through firewalls, ensuring privacy and protection from intrusion. Within the intranet, security features can ensure that information is distributed only to the appropriate people.”

Why should an office invest in the time to use the WWW to distribute information? Bernard (1996) discusses some typical problems that arise in large institutions:

- Incompatible, proprietary file formats
- Expensive, non-intuitive or even nonexistent viewing tools
- Frequent upgrading of publishing and viewing tools
- Massive printing budgets for documents that are under-used
- Out-of-date information locked in obsolete systems
- Difficult access to vital business information
- Redundancy and duplication of information across networks

---

*A browser is a program that allows a user to read hypertext by giving some means of viewing the contents of network nodes and of navigating from one node to another. Examples of browsers for the World Wide Web include Mosaic, Lynx, and Netscape (Pipkin, p. 45).*
Many of these same complaints have been raised at KU and surely at other universities, i.e., ‘Do we really need to have 400 pages of computer printout every fall semester when we only use about 50 of those pages?’

What the WWW and the Internet provide is a system to format and present documents using a typesetting language called Hypertext markup language, or HTML. As described by Pipkin (p. 46), “HTML is a hypertext document format used by the World Wide Web, built on SGML (standard general markup language). ‘Tags’ are embedded into the document text that allow links to other documents.” HTML is a rich language in its own right with many sources of information (A beginners guide to HTML found at <http://www.ncsa.uiuc.edu/General/Internet/WWW/HTMLPrimer.html> and a source for changes and updates in HTML can be found at <http://www.w3.org/pub/WWW/MarkUp/MarkUp.html>).

Aside from the ability to present information using HTML, the application development environment of the Web is rich. Jacobson offers the following reasons for considering the Web in application development at educational institutions (1995, p. 12):

- **Multi-platform:** Web clients exist for DOS, Windows, Mac, UNIX, and other popular operating systems.
- **Low cost:** Commercial Web browsers are available to educational institutions at no cost.
- **GUI:** Web applications may be simultaneously GUI- and character-based, delivering functionality to users of older desktop hardware.
- **Mixed media:** Web protocols support images, sounds, and video clips as well as text, allowing character-based administrative data to be merged with these rich data types.
- **Common user interface:** Although Web browsers run on disparate platforms, a certain look and feel is maintained across platforms, providing an easy-to-support common user interface.
- **Software distribution:** Web browsers themselves may be easily and inexpensively distributed across the network, using the Web itself.
- **Self-documenting:** Hypertext capabilities allow application help and tutorial routines to become an integral part of any Web application.
- **Distributed servers:** Web browsers merge information from several servers onto a single screen, without specific user knowledge of these servers.
- **Network security:** Socket-level encryption provides a secure network communications channel that can be employed to protect any existing or emerging campus authentication scheme in addition to all user data.
- **Local processing:** As Web browsers employ “helper applications” to display and process information, Web applications can therefore make use of local processes, such as spreadsheet or word processing programs.

But there is the issue of security. One of the disadvantages as well as advantages of the WWW is that it is usually accessible by anyone who wants to get to a specific HTML page. How can one restrict access? There are many ways, one of which has been previously mentioned, by an intranet. In addition, most WWW software that responds to requests for information, the http daemon, have built into them password protection capabilities. The how-tos, and pros and cons of such systems are found elsewhere on the web itself (CGI Security Tutorial <http://www.cscllab.uwaterloo.ca/~umlvanbie/cgisee/> and The WWW Security FAQ <http://www-genome.wi.mit.edu/WWW/faqs/www-security-faq.html>). It is safe to say, that few, if any, systems have foolproof security, and that a determined ‘hacker’ can find a way in. At the same time, after weighing costs and benefits, instructional institutions are probably not going to suffer dire consequences if someone gets in through a ‘backdoor’ somewhere and finds out the change in credit hours from one year to the next. What many deans or chairs want to avoid is for someone to be collecting information out-of-context.

At the suggestions of a group of associate deans who previewed the system, the WWW pages were placed under a password protection system to restrict access into what we call our Departmental Executive Management Information System, or DEMIS. Through this password protected system managerial information can now be
presented that is easy to reach, use, and discuss, while at the same time is hopefully enough of a deterrent for the causal, non-authorized users.

**How DEMIS Started**

KU’s institutional research office had in the past been responsible for putting together and distributing what was called *Departmental Profiles*. These reports contained dean and chair level management information concerning teaching loads, faculty salaries, enrollments, etc., for the past five to ten years on each academic department. There was a high cost to the office in the amount of work and resources that went into producing the reports, and there was doubt that the reports were even being used by many individuals on campus. While reviewing how the information could be provided in another manner, two things happened: (1) the office was connected to the Ethernet backbone on the campus, and (2) the Web explosion started to occur. Almost over night the means of providing access to departmental information to the campus was available — but where to start?

The computer center was contacted and a computer account on the campus Web server (UNIX platform) was obtained. After reading up on HTML, learning some about UNIX (the office’s first exposure), and mocking up some information pages, the conclusion was reached that *Departmental Profiles* information could be provided on the Web, but the same amount of work, if not more, would be required if the HTML pages were created using standard Web page authoring software. It was then decided to explore creating the HTML pages directly using the analytical reporting software, SAS.

SAS is currently positioning itself as a major player in data warehousing and providing institutional information via intranets and Web forms (for more information, see their home page at <http://www.sas.com>). Because the office had been using SAS on various systems, from an IBM mainframe, to Windows, and now to UNIX systems; many data bases used by the office existed as SAS datasets. It was soon found that with a little forethought and planning, these datasets could be used to write HTML pages as well as Excel spreadsheets in one pass per department from the analytical data sets.

An example of how a standard report procedure available within SAS (e.g., PROC TABULATE) can be wrapped within HTML code will be discussed next. What the user receives on his or her screen is mainframe output that actually looks somewhat presentable. An existing SAS program for doing a report on student level, sex, and ethnicity by major code was modified to created output files for each major at the university with the necessary HTML codes wrapped before and after the procedure output. The user can then print this report if he or she needs a hard copy of their distributions or examine additional major code distributions to get a picture of the majors of interest. The program creates a WWW area menu page, which has some valuable information of its own, as well as the major’s detail page. Figures 1 and 2 display how these pages are viewed by Netscape. The area menu contains HTML anchor tags, or jumps, to the major’s distribution detail page. If the user clicks on the underlined English BA (0185) anchor seen in Figure 1, the distribution detail by student level, gender, and

![Figure 1: WWW Menu for English Majors as displayed by Netscape](image-url)
ethnicity is displayed, as shown in Figure 2.

During this same time, SAS programs had been written to write data into Excel spreadsheets using the DDE (dynamic data exchange) capabilities of SAS for Windows. This was used to create what was called Statistical Overviews. The program wrote HTML pages that contained historical information on various categories of interest that could be viewed on the Web as well as Excel spreadsheets of the same data through the same pass of the data. This allowed those individuals who wished to further analyze their data the opportunity to retrieve from the WWW server with a click of a mouse the Excel spreadsheet. These files were then placed on a Web server under the DEMIS security access and made available to academic administrators. A presentation of the system was given at an associate deans meeting along with a survey which requested what additional management information the deans might be needing in the future. The results of the presentation were mixed. Some individuals were quick to look at the system for gathering additional information to use in their planning and budgeting process. Others were rather wary of why such information would be made available.

The main menu screen for the DEMIS statistical overviews is a simple page of all the schools in the university with the divisions and/or departments within the school following as can be seen in Figure 3. Once a particular school, division, or department is selected, the HTML version of the statistical overview is displayed. The user can scroll down and follow the various changes that have occurred on various measures for that particular unit. An index centered at 100% for fiscal year
1991/fall semester 1990 for the total sums of a particular measure was included. This year was selected because it was the middle of the time period being displayed. As additional years are added, the column would be recentered and the comparison index would change. The index can be compared with the indexes from other measures to see if credit hours are changing more rapidly than major counts, etc. Figure 4 illustrates how this page looks from Netscape.

Toward the bottom of the departmental overview are additional links that the user can click on to have the spreadsheet version of the report sent to their local PC. If the user configures the WWW browser correctly, the spreadsheet application can be started automatically upon clicking the HTML link. This allows the user to insert additional lines into the report and create different trend lines. These trend lines may be specific to undergraduate trends, tenure faculty workloads, or SCH taught by GTAs, but whatever they are interested in following, they already have the data in a spreadsheet format — the same data that will be used by the Provost in their budget planning sessions. Figure 5 is how the Excel spreadsheet looks when one clicks on the spreadsheet link.
After programming for various HTML pages, a few basic rules-of-thumb were found useful:

- **Use of information in database to create file and HTML anchor names**
  Make sure that there are enough acronyms or codes in the data to use macros, report breaks, labeling, and file naming for the HTML pages. For example, a one digit code for the schools works better than a complete school name to create files on a PC limited to eight characters for file names.

- **Organization and order of the data base becomes critical — more data records are better than more data fields**
  Codes that signal a change in what the data represent become extremely useful in programming for these types of systems. Instead of naming fields that represent SCH (student credit hours) for each year SCH94, SCH95, SCH96, etc., name it SCH and include in the data a record for each year and a code to represent the fiscal year. This allows the code to be used each year without changing the field names. In addition, the same block of code can be used with SCH, majors, or degrees with the only change being made in the code being: SCH→MAJOR→DEGREE.

- **Keep up to Date on Changes and Enhancements of Your Software**
  Being able to use SAS to write system commands and data directly into an Excel spreadsheet via dynamic data exchange (DDE) allows over sixty unique spreadsheets to be created in about an hour. A preformatted spreadsheet that acts as a template for the data is opened at each change in department. It contains the formats for the titles, centering, bolding, and italics. SAS then writes out the data into the appropriate areas of the spreadsheet and then sends Excel macro commands to name and save the spreadsheet in two formats, one in Excel and one in SYLK (for Macintosh users). A little time spent learning some new techniques such as this frees up a larger amount of time that would have been spent reading in delimited data and saving by hand.

**Development of Ad Hoc WWW Query Pages**

The next venture of the developing system came with some major changes in how KU was doing business. As had previously discussed, changes in how students were being charged for the courses they were taking created questions with major implications at the university. It was not known how students would adjust their course taking patterns nor which students would be the most likely to take additional courses or fewer courses.

The institutional research office was being asked about credit hours production weekly from various constituencies — primarily the Provost area but with various deans thrown in for good measure. What this meant for the office was that information needed to be provided to individuals whose questions often changed once they saw the data. While all these changes on campus were happening, the ability to use SAS as an ad hoc WWW report generator was being explored (Hoyle, 1996, Hoyle, 1994; “The SAS® System in Cyberspace,” 1996) and some demonstration sites were examined (http://www.sas.com/solutions/web/index.html).

HTML contains within it the ability to create WWW forms that allow the user to click on choices, select options, enter data, and then submit the selections and input to a program. Figure 6 is a screen snapshot of the DEMIS Credit Hours Ad Hoc Query form. Once the user makes his or her selections and presses the **Submit** button, the parameters selected on the form are passed to a program called a CGI script, that is actually located on another UNIX machine than
the Web form HTML page. This UNIX machine is primarily used for research and has as part of its system applications SAS. The CGI script parses the data stream being sent to it, writes a few lines of SAS programming code, and starts SAS to run the program. Once again, the WWW provided many sources that cover CGI scripting for those interested (one such place to start is Introduction to CGI Scripting <http://www.intergalact.com/hp/part3/part3.html>).

The way that CGI scripts are set to run on KU’s WWW servers allows the code to run using the system permissions of the account where the CGI script is located. The DEMIS CGI script takes advantage of this to call a SAS macro in the launching account which includes in it SAS system options that are specific for the analysis about to be done. This allows the same CGI script to be used with other DEMIS WWW Ad Hoc Query forms which would have different needs — only one parameter needs to be changed, the SAS program to be executed. The resulting SAS output and how it is displayed on a WWW browser is shown in Figure 7.

Response to the Initial Query Page
The response from ‘trial’ users has been very favorable. Each week new data is appended into the data set. Users could check any changes that had occurred in the past week, as well as compare the current week’s data with a previous semester for use as a baseline. The user could also go back in time for a previous week’s enrollment all the way to the first week of courses. Report breaks that can be selected are: (1) restrict data to specific schools or divisions, and then break out the data by (2) school, (3) department, (4) residency status of the students, (5) undergraduate/graduate course levels, (6) location of course, and (7) course number level. The ease of use and speed of this system has created a hopeful atmosphere among many academic administrators. In particular, a deans’ group has come together to meet with the institutional research office to discuss information that the deans would like to be able to query and have reported back to them.
To help facilitate the discussion with the deans, information about the university’s data systems was provided in a manner that was not overly technical, and yet at the same time which allowed them to conceptualize the types of analyses that could help fulfill their information needs. Grids were developed using the nine categories of information that Creswell and England (1994) described as being useful to chairs or deans. Table 1 displays a grid used for discussing data from KU’s student record system with the deans.

<table>
<thead>
<tr>
<th>Grouping Variables</th>
<th>Analysis Variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sex</td>
<td>Count</td>
</tr>
<tr>
<td>Ethnicity</td>
<td>Current credit hours enrolled</td>
</tr>
<tr>
<td>School Code (2 possible)</td>
<td>Cumulative credit hours A-D</td>
</tr>
<tr>
<td>Graduate Division Code (2 possible)</td>
<td>Cumulative credit hours A-F</td>
</tr>
<tr>
<td>Student Level</td>
<td>Transfer credit hours</td>
</tr>
<tr>
<td>Residency Status</td>
<td>Cumulative grade points</td>
</tr>
<tr>
<td>Major Codes (3 possible)</td>
<td>ACT scores - Composite, English, Math, etc.</td>
</tr>
<tr>
<td>Degree associated with Major (3 possible)</td>
<td>Age</td>
</tr>
<tr>
<td>Kansas Home County</td>
<td></td>
</tr>
<tr>
<td>Home State/Country</td>
<td></td>
</tr>
<tr>
<td>Marital Status</td>
<td></td>
</tr>
<tr>
<td>Housing Type</td>
<td></td>
</tr>
<tr>
<td>Commuter</td>
<td></td>
</tr>
<tr>
<td>Enhanced/created by OIRP</td>
<td>Variables that potentially could be added</td>
</tr>
<tr>
<td>Program of Major code</td>
<td>GRE scores</td>
</tr>
<tr>
<td>Department of Major code</td>
<td>Other standardized tests (MSAT, LSAT, etc.)</td>
</tr>
<tr>
<td>School of Major code</td>
<td></td>
</tr>
<tr>
<td>Age - by ranges</td>
<td></td>
</tr>
<tr>
<td>Campus - Official Board of Regents (No Edwards Info: Lawrence, KUMC, Off-campus)</td>
<td></td>
</tr>
<tr>
<td>Campus - Unofficial (Where majority of SCH is taken: Lawrence, KUMC, Edwards, Capital Complex, Other off-campus)</td>
<td></td>
</tr>
<tr>
<td>First-time freshmen</td>
<td></td>
</tr>
<tr>
<td>New transfers - from 2 year institutions</td>
<td></td>
</tr>
<tr>
<td>New transfers - from 4 year institutions</td>
<td></td>
</tr>
<tr>
<td>Continuing students by level (freshmen, sophomore, etc.)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Primary Statistical Analyses available: Sum, Count, Mean, Standard Deviation, Median, Mode</td>
</tr>
</tbody>
</table>

The deans group has been very useful in directing the development of the system. At this time, they are still going over the information grids that were provided and plan to reach a consensus in prioritizing the development of DEMIS in the spring of 1997. At the same time, the office of institutional research is holding training sessions with associate and assistant deans as well as department chairs to give them exposure to the system and how to optimize some of Netscape’s options for DEMIS. These training sessions usually last between sixty and ninety minutes and have gone very well — especially since most already have had experience using a Web browser.
Conclusion
At this time, DEMIS is in a state of continuing development. After receiving the priority list from the deans, additional types of queries will be developed based on their order on the list. The training sessions being conducted this spring provide constant feedback and a level of anticipation and expectations has developed. Four gigabytes of additional disk storage for use on the UNIX machine have been set aside for developing the SAS data sets used in DEMIS so there should be no problem in creating the analytical data warehouses for the queries. The university administration is further supporting the development of the system by providing the office of institutional research with an additional part-time programming position because the need for information in today’s educational environment is so crucial. As Keller (1995, p. 64) puts it, “...those in charge will need to have excellent information on the operations within their campuses, on emerging issues and developing trends, and on quality indicators. Superior academic management and outstanding institutional research must be more closely tied than ever before.” Hopefully, a step in that direction has been made.
References


SAS is a registered trademark of the SAS Institute Inc. in the USA and other countries. ® indicates a USA registration.