

Software Development in Higher Education

L. Russell Watjen

Educators in general, and higher education professionals in particular, are experiencing the results of a failure of computer software development to keep pace with hardware acquisition and development. A few short years ago there was very little computer hardware available to educators and student service professionals. Where hardware was available, few people knew what to do with it. Since that time, computer manufacturers, anxious to make sure that the next generation of college graduates are knowledgeable not only about computers but about their brand of computer, have provided attractive purchase discounts. Together with numerous lay advocates, they have convinced educators of the necessity to promote computer literacy. Unfortunately, availability of well-written and documented software has not kept pace with hardware distribution.

Today, microcomputers are available to significant numbers of educators not only in the workplace but for personal acquisition, often at attractive discounts. With this availability, a new era in the application of computer technology has come to the campus. Thanks to numerous individuals and organizations that have sponsored so-called computer literacy programs, there is now a substantial number of educators with a basic knowledge of computer capabilities who have begun to utilize newly acquired equipment. Educators have begun to move from dependency on centralized computer systems to reliance upon decentralized and stand-alone microcomputer

L. Russell Watjen is Vice President for Student Affairs, Wentworth Institute of Technology, Boston, MA.

applications as well. With these changes, new challenges have arisen not unlike those faced by business and industry in previous years. Without proper software, the most advanced piece of hardware is little more than an expensive paperweight.

Currently, microcomputers are frequently used for word processing and with various types of general purpose software. They are also frequently used as dumb terminals. Most microcomputer applications to higher education are administrative and involve the use of general purpose software. The challenge now faced by those in higher education is to find ways to go beyond straightforward applications of microcomputers and move to more creative and innovative applications. Although powerful by comparison to what preceded them, in many cases current microcomputer applications to higher education have not lived up to their full potential. The real challenge, and the greatest potential for realization of this potential, lies in the development of special purpose applications software. Computer simulations, decisions support systems, and innovative communications applications are but three areas that have barely been touched in higher education.

Substantial obstacles must be overcome before higher education can move on to the next stage of computer-assisted delivery of student services. There continues to be an aversion to the use of microcomputers on the part of some individuals, particularly those in the helping professions. Others who would use microcomputers lack the requisite skills or equipment. Still others with the requisite skills and equipment lack the necessary time to use their skills in designing appropriate computer applications. Another factor has been the lack of communication between campus computer specialists and specialists in various higher education and student service areas. Perhaps the most significant factor, however, is largely external to the field--the economics of software development does not provide great incentive to its development in higher education.

The aversion to microcomputer use is not unique to student services professionals. This phenomenon, however, appears to be somewhat stronger in the helping professions than in other fields. While there is no easy solution to this problem, perhaps hesitant professionals will begin to use microcomputers when innovative and creative applications emerge.

Provision of needed skills to those who had no exposure to microcomputers during the course of their professional preparation is well underway. Workshops and short courses have been offered by organizations such as ERIC/CAPS, by local and national professional organizations, and by institutions of

higher education themselves. Little can be done, however, to make more time available to educators for development of computer applications to higher education. Ultimately, each person must decide whether the cost in time and energy taken away from other activities is justified by the advantages gained. Very often, when viewed from the perspective of one campus in isolation, the answer to this question is no.

Steven W. Gilbert, managing director of EDUCOM, a higher education computer consortium, describes the lack of computer software for higher education as a problem "only dimly understood." He suggests that part of the problem is that "those who develop software and those who merely want to use it are unaccustomed to speaking to one another" (Turner, 1985, p. 26). There is some evidence that this is becoming less true.

As noted earlier, perhaps the most significant influence on the development of specialized student services software is the lack of economic incentives. In order to fully understand the nature of these incentives, one must understand more about the nature of software markets. Computer software markets are described as "horizontal" or "vertical" markets. A horizontal market for a product is one where the same piece of software is applied to numerous, similar applications across a field or several fields. Accounting and inventory programs lend themselves well to horizontal markets. One accounting package might be useful for budget development by a career development center as well as for nearly every other office in an institution. This horizontal applicability is true also of many general purpose software application packages, such as databases and spreadsheets.

A vertical software market is one in which software has been developed for a narrow and specific application. This software might, for example, assist students to learn how to develop resumes. As such, the program might lend itself extremely well to career development and placement centers in general, but have very little or no applicability beyond these areas. Specialized room-assigned programs, student activity scheduling programs, values clarification exercises, and computer-assisted guidance programs are but a few examples of software programs which have been developed primarily for vertical student services markets. Exploitation of such markets is necessary in order to make it economically attractive to spend the time and money necessary to produce high quality software. In higher education these markets are small in comparison, for example, to markets for medical applications software. How many interviewing scheduling programs or values clarification programs will the market support? Not many, in all likelihood, compared to the market for doctors' offices'

billing programs, etc. The medical software market also is likely to have less difficulty with funding systems.

The problem of developing and distributing software to vertical markets is not unique to student services. Any field where the number of consumers is relatively small will have difficulty achieving economies of scale so as to generate low prices for consumers and reasonable profits for authors. On top of this, as in other fields, higher education faces the problem of a vast array of hardware already in use and the difficulty and cost of modifying software to run on different machines.

Steven Burke (1985), in a recent article dealing with advantages of specialized software development, describes similar problems in business and industry. In industry, vertical markets for software are frequently being serviced by so-called "total solution" applications. Some software developers specialize in developing software for a particular market, package this software with a particular piece of hardware, and include installation and training in the sale price. In this way, problems of hardware compatibility and support are solved. Unfortunately, the "total solution" type of software delivery system is expensive and does not lend itself well to all applications. A computerized enrollment management computer system is another example of a vertical software market where the potential demand is sufficient to justify the cost of system development.

There is a finite number of things that can be done to expedite the rate of software development in higher education. Good software development is slow and, for the foreseeable future, the number of persons in the ranks of higher education who have the time and combinations of expertise necessary to develop needed software is small. In colleges and universities where computer support exists, the time and effort necessary to cultivate communication and cooperation with existing programming staff will, therefore, be time well spent. For many student service offices, the in-house computer center staff are the best short-range hope for obtaining software development assistance. Monetary incentives have not been strong for commercial software developers to work with student development specialists. Persons with the requisite skills often find far more lucrative employment working with other fields. With the recent downturn in high tech industries, perhaps sufficient numbers of persons with programming expertise will become available for employment in higher education. Those fortunate enough to have sufficient budgets may then be able to purchase their services.

Professional organizations have played and, one would hope, will continue to play a significant role in promoting computer literacy. What is needed are these organizations' continued support and sponsorship of efforts to promote communication among professionals about software and their encouragement of experimentation with software applications. Illustrative of these efforts is the ERIC/CAPS high tech electronic network. This service provides a convenient method for the electronic exchange of information between geographically dispersed professionals. Student services professionals can communicate with each other at whatever time is convenient for them and from wherever they happen to be. In the future, services such as this have potential to provide a medium for collection of information, such as software evaluations and the sharing of information regarding software availability and applications. SAMI, the Student Affairs Management Information system being developed by NASPA and ACPA, is an administrative example of professional organizations collaborating to develop innovative software applications. Additional information regarding the ERIC/CAPS network (CAPS:HITECH) may be obtained by writing the ERIC Counseling and Personnel Services Clearinghouse, 2108 School of Education, The University of Michigan, Ann Arbor, MI 48109-1259. Information regarding SAMI may be obtained by writing the National Association of Student Personnel Administrators, P.O. Box 21265, Columbus, OH 43221.

A major challenge remains: finding ways to foster the development and distribution of high quality, special purpose software. Although an important hurdle was cleared with the acquisition of necessary hardware by many institutions, there is still the lack of economic incentives. Research grants by professional organizations and mechanisms similar to those used to stimulate development of medical software would be important advances in addressing this problem. Until such incentives are offered, software development will likely continue at a slow rate.

REFERENCES

- Burke, S. (1985, July 15). Specialized packages help increase profits. Infoworld, 7(28), 30.
- Turner, J. A. (1985, September 4). Campus computer use expected to soar this year as "early follower" colleges try to catch pioneers. The Chronicle of Higher Education, 26.