ASPECTS OF COMPUTER EDUCATION AND S/390

David C. Thewlis

DCTA, Inc.

May 1999

Introduction

This paper discusses computer education and especially computer science curricula as they relate to large-scale commercial computing (e.g. the IBM S/390 Enterprise Server), and suggests areas where substantial improvement seems to be necessary. The underlying premise is that computer science curricula and other computer and information technology offerings available via academic institutions today are largely theoretical and are focused on desktop or confined computing models. Such computer science courses are intended to provide the student with the educational foundation to pursue graduate work in the discipline of computer science, generally with a view towards a career either in academia or in research and development work. Even in the collateral courses offered in MIS (management information sciences) or CIS (computer information sciences) courses, little or no attention is paid to applied programming, large-scale programming practices, or the notion of programming and information systems management as a shared discipline.

Unfortunately, the vast majority of jobs in the disciplines of information technology fall into precisely those areas where little attention is paid, and traditional computer science graduates, or MIS/CIS program graduates, are not prepared to enter the job force without substantial investment on the part of their prospective employers to give them the skills and specific education they need to be effective professionals. While this is true to some extent across all areas of information technology, it is particularly serious in the field of commercial and large-scale enterprise computing.

Most academic institutions take the position that it is their primary goal to provide to their students an educational foundation upon which the graduate may build any necessary specific skills, and downplay or even reject providing specific job-related skills. While providing the educational foundation is laudable and correct, it is insufficient in today's world where employers increasingly are not willing to invest time or money in entry-level employees, but wish to hire people who have the precise skill sets they need and can immediately be productive.

This focus on theory is partly due to academia's being primarily aware of its own need to further the state of computer science and the teaching of computer science, but it also derives from a lack of what might be called a computer or information technology "engineering" focus. In most scientific fields both engineers (practitioners) and scientists (researchers) are needed, and education towards both disciplines is provided. Unfortunately, in information technology, while the “science” part is provided for, the "engineering" disciplines (not "software engineering" or “computer engineering” as generally stated, but a practitioner focus on the use of computing) is largely ignored.

The view expressed in this paper is that the lack of a formal education for what might be termed "information technology engineering" is at the heart of a dilemma facing students and employers alike. Many (possibly most) students seeking a career in the fields of information technology enroll in computer science curricula because a curriculum with the needed focus simply isn't available -- yet the purpose and focus of the existing computer science curricula appears to be largely to prepare eventual scientists and doctoral candidates. The vast majority of graduates enter the work force rather than pursuing an advanced degree, with little or no grounding in the general disciplines which underlie most commercial computing, in addition to the theoretical grounding already being taught.

In addition to the issues raised by the nature and focus of computer science and information education today, a particular problem arises with respect to both the education of managers in information technology fields, and the study of the management of information technology, particularly in the large-scale commercial computing, and these related topics are explored in some depth.

Aspects of Education

From the perspective of this paper, information technology can be largely divided into three general areas:

1. Practitioner development: programmers, systems analysts, systems developers, network technicians, and the array of staff working for an organization using computers, either large-scale or otherwise; together with the development of knowledge about the practice of computer programming.

2. Management development: the development of IT management and executives at all levels and the development of knowledge about the management of information technology.

3. Academic: the development of academicians and scientists who will advance the state of the art in academia, research or business, and the development of pedagogy in the field. Related to this, but not directly a part, of it is the work of computer scientists whether in academia or elsewhere advancing the state of the art.

Please note that these areas have less to do with the “level” of education (undergraduate, graduate, etc.) or the source of the education, than with the purpose to which the education is put by the graduate.

Practitioner Development

1. Practitioner development: programmers, systems analysts, systems developers, network technicians, and the array of staff working for organizations who use computers, either large-scale or otherwise; together with the development of knowledge about the practice of computer programming.

Education in computer science for eventual scientists and academicians is obtained through undergraduate and postgraduate computer science programs. Most such programs offer a good grounding in theory, architecture, and the various disciplines the future scientist will require. They are not focused upon the practitioner in the sense that this paper uses the term.

By contrast, practitioner development, that is, developing professionals who actually do the things related to the use of computers, occurs in a variety of ways, many of which are not part of formal education or which are not generally provided by academic institutions, at least at the degree-offering level. As a result, much of this is learned after the fact or by specific offerings at separate training institutions. Practitioner development can be divided into:

- Formal education
- Skills training
- On the job training or development

Formal education for practitioners often comes through existing computer science programs, which at least at the undergraduate level are not wrong for practitioners although they seem to be insufficient, and certainly do not graduate enough students to satisfy the needs of the IT industry with respect to software developers and other IT professionals. However, many practitioners enter the industry via the more skills-oriented offerings of two-year institutions, through training courses established by companies, job-transfer programs at the local, state or regional level, or by transferring into an IT organization or assuming part-time duties such as LAN administration in their organization.

Employers use a variety of techniques to develop their employees, ranging from vendor-supplied specific skills or technologies education to simply having the novice work with an established practitioner and learning by example. Some large firms have established programs with local colleges and universities, or use a separate training institution to provide initial startup training. In general, however, unless established professionals are hired who already have the education and skills needed, a substantial investment in time and resources is required for new employees.

Clearly, the development of "IT Engineering" curricula and offerings which focus both on specific technologies and on the disciplinary approaches to shared program development and production-level systems administration and management could greatly alleviate the need for initial compensatory education for recent computer science graduates, and would appear to offer a computer science/engineering student a way to become proficient in needed disciplines, and possibly specific languages or technologies, if he or she intended to enter the workforce upon graduation.

Management Development

2. Management development: the development of IT management and executives at all levels and the development of knowledge about the management of information technology.

This subject covers both the development of IT managers and executives, and the development of knowledge about managing information technology. Some universities offer information technology courses via their business schools, separate "information technology" institutes, or curricula in Management Information Science or Computer Information Science. These frequently have been implemented without the cooperation or involvement of the computer science department, which does not usually see such "business oriented" courses as part of its domain and frequently does not regard them as education at all. These courses may not be part of a degree program (or are offered as adjunct rather than core courses), and are often in fact offered as independent continuing education courses to those already in the workforce.

A slight variation on this theme is courses offered as part of a specific program such as a Master's in Business Administration with a focus on Information Science: in these cases the IT-related courses usually form what might be called a "minor" in an undergraduate program, but generally are not the substantive major component of the MBA program. There are also a few schools who offer such a program where the IT component is in fact the major focus.

Unfortunately, although the focus of MBA programs is almost universally that of management training, few of the IT-related courses in these programs or other business school offerings actually have much to do with the management of information technology, and almost nothing with the management of large-scale commercial computing centers. Instead, what is often offered is a series of "overview" courses designed to give the potential MBA graduate some concept of the technologies and issues.

In fact, there is a tremendous lack of actual management education in the field of information technology and commercial computing. This is true not only for universities and colleges but indeed for offerings from specialized institutes and vendor-supplied education. Where courses are offered in areas such as systems management tools, they are generally aimed at the practitioner whose job it is to use the tool, rather than the manager whose job it is to understand the issues and make decisions about the IT environment.

This has really been true for many years, and as a consequence managers in information technology organizations either are promoted up from practitioners and have to develop both management skills and management perspective on the job (not always successfully), or they are skilled managers put into information technology departments on the theory that a good manager can manage anything -- perhaps true in theory, but frequently disastrously insufficient in practice.

While this problem affects all organizations with information technology groups or sections, it is particularly traumatic for large organizations with centralized IT organizations and data centers. An understanding of the systems and applications provided by the central organization is essential to make informed and appropriate decisions on problem and change management, allocation of resources, growth and capacity, to name only a few issues. This understanding requires a broad grasp of both the technology areas and the business areas, and their capabilities, needs, and limitations.

However, as the general understanding of computing becomes more and more dominated by the workstation, the LAN, and the UNIX Server, management candidates in common with IT practitioners do not have a comprehension of the necessary disciplines to implement and conduct a serious, cost-efficient organization, nor the empirical knowledge of these techniques which has developed in the business community over time.

There is a great need for the development and offering of a cohesive program on implementing and managing a large-scale computing environment, covering the technologies, the management disciplines, the capabilities of the systems management tools, and how to put together and successfully run such an environment. The courses in such a program should be credit courses and part of a degree program, and available both within the program and externally, for undergraduate or graduate students interested in a focus on large-scale computing management.
Many of the components of such a program (perhaps leading to a Master of Business Computing Environments degree or some such) could be traditional courses offered in a business school (even as they are today) to put together a complete degree program, but the focus and IT-related offerings must be constructed, as they do not seem to exist today.

Indeed, such courses should be offered to business personnel interested in becoming high-level managers and executives in the IT field, perhaps on a noncredit or specially organized basis. Many business schools offer a variety of specially-tailored programs to support the working manager including evening courses, weekend seminars, and different structures built upon the comparative discipline (and lack of time) of the working professional as opposed to the full-time student.

The other issue in management development is the development of knowledge about managing information technology itself, which may be necessary to develop the necessary management courses. Simply stated, the knowledge of real world, large-scale computing that exists today is largely if not completely empirical. Studies of issues as diverse as distributed data, multinational scheduling and calendaring, dynamic backup and recovery, and many others, have almost never been done using formal or academic disciplines, except for very small controlled models.

Companies who do these sorts of things, and companies that develop the hardware and software to manage and control highly-available, very-large-scale commercial computing environments, generally do so with an empirical (and often very detailed) understanding of the problems. What is frequently lacking is the rigor of a study as to formal comprehension and elegant solutions, and the software and hardware tools are consequently very often overly complex and unsatisfactory.

An interesting irony is the ongoing expansion of the uses of information technology (as opposed to the frontiers of the technology itself). Computing and IT is becoming pervasive in commerce, society, and indeed the course of life. The capabilities being built on the Internet, including such things as electronic commerce, online trading and shopping, collaborative working, virtual reality, and so forth are rapidly moving from state-of-the-art to being integrated into all parts of our culture and society.

As this happens, the need for the rigorous disciplines ranging from shared software development to 24x7 operation, in-flight software upgrades and business recovery become more and more necessary.   The large-scale enterprise server systems such as S/390 are evolving to offer the technologies and capabilities our industry is developing and our culture is absorbing. In the process, they combine these technologies with scalability, availability, security, change management and capacity not otherwise available, and are being adopted for use in these innovative areas by organizations requiring both innovative technologies and responsible management.

But these techniques and technologies, although increasingly important, remain largely empirical in nature. In general, there do not seem to have been any (or not many) serious efforts on the part of computer scientists or academia to examine the sorts of problems faced by large-scale commercial enterprises in using computing systems. From the academic perspective it may be that such problems as a class do not appear to be interesting. It is also at least as likely that they are not seen as credible projects or areas of investigation for postgraduate studies. Whatever the reason up to now, the empirical knowledge won by trial and error needs to be augmented by formal study of the issues surrounding large-scale commercial computing and the development and implementation of very large application systems in these environments, and academia and academics must be encouraged to explore these subjects.

Academic and Knowledge Development

*3.	Academic and knowledge development: the development of academicians and scientists who will advance the state of the art in academia, research or business, and the development of pedagogy in the field. Related to this but not directly a part of it is the actual work of computer scientists whether in academia or elsewhere advancing the state of the art.*

In many ways this is the most robust of the three areas being discussed, as computer science programs are largely devoted to the development of computer scientist doctoral candidates who will become research scientists and/or lecturers and will both further the state of the art itself, and develop the pedagogy of teaching computer science. There are many estimable universities and colleges with excellent computer science programs from this perspective, and they fit fully into the academic model.

Unfortunately, as touched upon above, the perception that computer science education should be strictly theoretical and not applied, or affected by the real world of programming practices and commercial computing is very real, and very widespread. Whether the "right" answer is the development of "computer engineering" or “information technology engineering” as suggested above, with the same foundation courses and parallel but different courses at a higher level is unclear; this paper uses the concept simply to point out the problem in a familiar context. But it is one that needs to be addressed to allow the many computer science students who are interested in entering the job force after graduation to acquire an education more focused on the problems and issues they must master.

Strictly speaking, the references to “the development of knowledge about the practice of computer programming” and “the development of knowledge about the management of information technology” discussed under the sections on practitioners and management above, are part of the field of academic research and investigation; however they were brought up and briefly discussed in those sections as they logically fit there.

Wherever discussed, however, there is a need for computer science programs and computer science academicians to take commercial computing seriously from the perspective of academic investigation and the furtherance of knowledge and formal studies on many of these issues. Textbooks on (for example) the organization and implementation of multiprocessing operating systems completely ignore the OS/390 operating system and its predecessors, even though those are the most robust and mature examples of such systems. Similar issues exist across the board with respect to both textbooks and academic research into computing technology and phenomena.

Part of this issue, of course, is often discussed in the context that researchers are hampered by lack of access to such systems in many academic institutions, and consequently will research what they either know or have available, to say nothing of teaching the same things. This is certainly a real problem, exacerbated by the relatively high cost of these systems and the tendency to want "one's own system" for research or teaching.

On the other hand, in many of those institutions, such systems do exist -- but in an administrative data center rather than in the computer science lab. While lack of access may still be a problem (and the notion of letting the students loose on the administrative computing facility is unlikely to find much support among the administrators), it is also probable that the access issue could be resolved if necessary. Instead, there is another and more pervasive issue: the general perception that these systems are dated and not appropriate for academic research or even academic use. While the reasons for this perception (if real) are certainly myriad, how we have gotten to this point is less important than what we should do in the future.

In Conclusion

Neither academia nor business can afford a biased and incomplete view of the tools and technologies available to solve the real computing issues in the world today and tomorrow, nor the adequate preparation of students to enter that world. It is time to adopt a balanced view of the major systems and technologies available, from the perspective of the appropriate tool to solve the problem, as opposed to near-religious bias towards one or another architecture, implementation or language as being appropriate to all purposes. Correspondingly, it is time for computer science and computer science education to adopt the same perspective, develop curricula which adequately cover the real breadth of information technology, and educate students in the context of what is actually needed in the world of computing both today and tomorrow. And finally, it is time for the legitimate areas of study and research in large-scale commercial computing, which up to now have not been explored rigorously, to become part of the scope of academic study.