

AC 2007-1076: NON-TRADITIONAL BACHELOR DEGREE PROGRAMS AND OPTIONS OFFERED BY ENGINEERING SCHOOLS: THEIR IMPACT ON TRADITIONAL ENGINEERING PROGRAMS

E. Bernard White, George Mason University

E. Bernard White received the Doctor of Philosophy (Ph.D.) degree from the University of Virginia (Charlottesville) in Systems Engineering. He has studied at a wide range of universities, earning bachelor and master degrees in mathematics from Mississippi Industrial College and the University of Illinois (Urbana), respectively. He earned a master degree in Urban Systems Engineering from Howard University (Washington, D.C.). Mr. White worked as a research engineer and assistant professor in the Howard University School of Engineering prior to working as a Systems Engineer at the MITRE Corporation (McLean, VA). Mr. White is currently Associate Dean for Undergraduate Studies and Associate Professor of Systems Engineering and Operations Research at George Mason University (Fairfax, VA)

Non-traditional Bachelor Degree Programs and Options Offered by Engineering Schools: Their Impact on Traditional Engineering Programs

Abstract

Our Volgenau School of Information Technology and Engineering (IT&E) at George Mason University established a new four-year Bachelor of Science in Information Technology (BS-IT) program that focuses on selected sub-disciplines or areas of concentration during the last two years. The BS-IT program aims to help fulfill The Volgenau School's mission, which is to produce more graduates who understand information technology and who can use it effectively. The BS-IT degree was offered for the first time during the fall, 2002 semester. Enrollment in the BS-IT program at the beginning of the fall 2006 semester was over 900 students, making it the largest undergraduate program in The Volgenau School of IT&E. The Computer Science (CS) Department in The Volgenau School here at George Mason has developed a new Bachelor of Science degree in Applied Computer Science (BS-ACS) with concentrations associated with selected disciplines (e.g., Biology and Geography) beginning with the fall 2006 semester. The decision to offer the BS-ACS program was influenced partially by the success of the new BS-IT program and the declining enrollment in the traditional BS-CS program, as well as the perceived need in the computational sciences profession. The BS-ACS program proposes to retain stringent technical requirements in the CS major related courses as well as in the discipline associated with the selected concentration. It must be noted that much of the success of the BS-IT program can be attributed to its significantly reduced mathematics and programming requirements, which is not the case for the BS-ACS program. In this paper, we take a closer look at the types of non-traditional programs that are available to undergraduate students in engineering schools and explore the extent to which these types of non-traditional programs are achieving their goals both in The Volgenau School and at other select universities across the country. Additionally, we explore the impact of these types of non-traditional bachelor degree programs offered by engineering schools on more traditional undergraduate program offerings in engineering and computer science. We also attempt to both identify and gain a better understanding of the full range of issues that must be addressed. Among the topics to be discussed are: issues associated with both the recruitment and persistence of students in the traditional engineering and computer science programs; need for new types of engineering and computing related programs; and the need for branding of the non-traditional programs so that they are easily distinguishable from similarly named traditional programs. The results should be of interest to engineering schools and other academic units that are contemplating and/or in the early stages of implementing non-traditional bachelor degree programs.

1. Introduction and Overview of Paper

Back in the 1960's, electrical engineering, computer science, and information systems were essentially the only computer-related undergraduate programs available to students.⁹ These three areas were well defined with little, if any, overlap of their respective domains. After the invention of the chip-based microprocessor in the 1970's, the field of computer engineering first began to emerge as an area of study. In the 1990's, both software engineering and information technology (IT) emerged as separate entities. Further refinements in the information systems field continue to occur as well; however, this paper focuses on changes that are more aligned

with the evolution of the traditional fields of electrical engineering and computer science, with emphasis on the area of computing.

As we approached the end of the 20th Century, problems like Y2K placed increased demands on the high technology community. This led to increased pressure on engineering schools to increase the production of professionals who have sufficient computing skills to meet their technical needs. Non-traditional programs like our BS-IT major are a direct outgrowth of this demand from industry. With the relaxation of mathematics and other technical requisites, we were able to attract and retain a greater pool of students who previously found undergraduate programs in engineering schools inaccessible. In this paper, we identify some of the major reasons for the decline of student interest in traditional engineering and computer science majors. We take a closer look at the perceived loss of jobs in the IT industry as well as some of its causes. We also examine the complexity of issues that are fueling the need for non-traditional programs as well as approaches that some institutions are taking to address the changing needs. We examine how the pool of students available for various types of majors is affected by program requirements for more traditional majors and examine the impact that non-traditional programs and their degree requirements are having on this pool. We conclude with a summary of some of the types of challenges and opportunities that will continue to present themselves to program designers and administrators.

2. Reasons for Declining Student Interest in Engineering and Computer Science Majors

Fewer high school graduates are selecting computer science and engineering majors for a number of reasons, including a lack of understanding and appreciation of curriculum requirements for these majors and resulting career opportunities associated with a specific major. Because there is such a plethora of subject matter in the fields of computer science and engineering today, it is not surprising that many students today are confused and are unable to easily identify a single major that covers all of their areas of interests sufficiently. Both high school counselors and parents from whom these prospective students seek career-related guidance are confounded as well.

Much of the declining interest in computing by the high school student can be attributed to the inability of the computer science professional to articulate its profession as little more than programming, in spite of all of the new knowledge and exciting high technology developments. In addition to programming, the computer science professional must be able to present its field more broadly as the discipline concerned with important areas of design, development, testing debugging, documentation, and maintenance of software. For example, the computer science professionals have failed to convey to prospective students and counselors the fact that computer science provides the capabilities that enable information technology applications used by the IT professional. Other factors that have contributed to the decline of interest in the traditional computer science major include the following:

- The misperception of severe job loss in the IT sector
- Media portrayals of computing as nerdy and boring compared to other fields
- An impression that computing requires extraordinary proficiency in mathematics
- Students and many counselors who view classes in mathematics and CS as too difficult or uninteresting

- Uninformed yet well-intentioned high school counselors
- An increased interest in fields such as the biological and professional sciences, including medicine.

The Volgenau School of IT&E has many non-traditional students for which a traditional computer science or engineering degree might not be well-suited. The BS-IT degree, however, would be well-suited for a greater percentage of these non-traditional students who tend to be older students returning to college for a degree, and they often have family responsibilities as well as part-time or even full-time jobs. Some of these students already have degrees in non-technical areas, and many are already working in high technology positions. Many have attended community colleges and other institutions, and some have earned IT certifications in a variety of areas.

Hoffman¹⁴ makes the observation that students arriving on college campuses today are increasingly sophisticated users of computing technology applications like word processing, email, and other Internet capabilities. Hoffman notes further that “The perceived diminished relevance of Computer Science is being driven by two complementary phenomena: computing applications are being viewed less as applications of computing technology and more as integral components of everyday tasks, and the field of computing is examining itself in an effort to understand how to prepare students for rapidly changing careers increasingly integrated with other disciplines.” These types of observations require that educational program administrators and other computing professionals continually review the evolving field of computing in an effort to understand how to prepare students for rapidly changing careers increasingly integrated with other disciplines.

3. Reasons for Loss of IT Jobs

It is true that the high technology sector has experienced some types of job losses; however, at the same time there have been increases in many other areas. A variety of factors have contributed to these losses in the high technology related occupations, including the following:

- End of work on Y2K and the dot-com bust
- Terrorist attacks of 9/11/01 and their related effect on the U.S. economy
- Downturn in corporate IT spending
- Brief 2001 recession
- Productivity increases as a result of technology
- Off-shoring of IT work.

Additional reasons for the loss of jobs that were traditionally reserved for the computer scientist can be directly attributed to the unprecedented accomplishments of the computer science professional.¹¹ Included among these accomplishments are the following:

- Development of easy-to-use interfaces
- Intuitive end-user programming environments requiring little training
- Embedded applications
- New hypertext markup and scripting languages
- Highly graphical languages such as Visual Basic

- Tools such as word-processors, spreadsheet, and presentation software
- Intranet and Internet communication capabilities.

To some it might appear that computer scientists have reduced the complexity of programming to the point where formal training in computer science is no longer needed.¹¹ As a result, the standing of computer science as a discipline is now questioned. For example, Internet, data transmission, and data processing capabilities have facilitated off-shoring. If computer science is to survive as a discipline, a systematic study of these types of problems and the underpinnings of such a revitalized discipline must be determined.

According to Cardoso⁴, pervasive computing, ubiquitous systems, wearable computing, and ambient intelligence are all new topics that will play an important role in future computing systems. These topics are all somewhat related to embedded systems. Smaller computing devices, such as Personal Digital Assistants (PDAs), tend to have the potential of recent personal computers (PCs) but with severe restrictions in memory size and energy consumption. Far too few of our computer science students know that the way a particular application is programmed has impact on the amount of time we can use a PDA running the application without recharging the battery. Many do not know, e.g., that a scrollbar in a PDA's GUI increases energy consumption.⁴ It is predicted that by the year 2010, 90% of the overall program code developed will be for embedded computing systems, and this is just one area where the computer science professional has failed to use common everyday devices to convey the importance of its profession.

Hoganson¹¹ has analyzed the effectiveness of outsourcing and concluded that IT-related business enterprise functions differ markedly in the ease and success in which they can be outsourced. Functions and knowledge areas that are of strategic importance to the enterprise and those that are critical to a firm's competitive advantage are less likely to be successfully outsourced. Similarly, Saad¹⁶ has determined that the nature of technical work in industrialized societies has been undergoing fundamental changes. These and similar observations can serve as the basis for overhauling the organization and content of the current computer science curriculum to increase the value of our computer science graduates, as advocated by a number of academics.⁴ For the novice, implementation of appropriate overhauls to address outsourcing related issues and changes in workforce requirements generally appear to be obvious; however, for the seasoned computing and engineering program administrators and educators who are more familiar with the nature and culture and of educational institutions, devising effective strategies for addressing the issues is becoming increasingly challenging.

4. Need for Non-traditional Programs

The computing profession itself has many knowledge areas each requiring specific areas of expertise. Among these are the following:

- Object-oriented programming
- Relational databases
- Networking technologies
- The Internet, World Wide Web, and its applications
- Graphics and multimedia

- Embedded systems
- Human-computer interaction related issues
- Application programmer interfaces
- Application domains
- Interoperability
- Cyber terrorism and information warfare
- Security and cryptography.

Additionally, the nature of companies seeking employees has changed dramatically over the past 20 years. There is an increasing need for “hybrid” workers who both understand technology and also know how to apply, deliver customer services for, and manage sales of information technology. An increasing number of professionals who do not have educational backgrounds in engineering or computer science are working in the information technology field. Cukier⁷ notes that “The convergence of telecommunication, computer hardware, software and “content” is evident in the emerging technology infrastructure (e.g., the Internet), the applications (e.g., Web design) and the industry itself (e.g., new multimedia corporations).” In addition to changes in requirements for entry into the technical workforce, many who enter the employment sector after receiving highly technical engineering or computer science degrees become managers early on in their career. These graduates entering the workforce with a more traditional engineering education are generally lacking in any component that educates them in how to deal with management issues.¹⁰

Also, there are increasing demands in industry, government, research and educational institutions for graduates who work as scientific application programmers.⁵ The scientific programmers’ positions are currently filled by people trained in a scientific discipline who have taken one or two programming courses or have learned programming through practice on the job. Generally computer science undergraduates are capable of taking on such jobs after some additional (or on-the-job) training in the specific scientific discipline.

As the nature of computing continues to change as well the introduction of new engineering technologies, the nature of businesses and their needs will continue to evolve as well. The inter-relations among the innovative high technology institutions and enterprising businesses comprise a dynamic loop or set of chain reactions that requires that educational programs in engineering and computing must continually review and revise courses, programs, and their curricula. Again, a comprehensive and systematic “evolution” of the computer science and engineering knowledge areas is required to develop an effective strategy to position the computer science curricula so that it can be more vital to the health of a company as well as more capable of responding appropriately to enterprise decision such as outsourcing that could adversely affect the careers of technical professionals.

5. Approaches to Meeting the Need

Since curricula for traditional computer science and engineering programs have been “stretched” to cover important concepts for years now, how does a program such as computer science accommodate the ever increasing number of knowledge areas and the new advances in technology? Clearly, a bachelor degree computer science curriculum cannot dedicate new courses to each new advance if the program wishes to maintain its accreditation by ensuring that

the majority of the Computing Curricula 1991 subject matter is still covered.¹⁷ Even when programs strive to be more open and agile, they are still challenged to incorporate emergent technologies. A preview of the technical literature reveals the following types of strategies that have been used by both computing and engineering departments to accommodate the explosion in technical knowledge and new technologies:

- Develop new courses that integrate as many of the new technologies as possible, while streamlining current curricula to allow room for these new topics.¹⁷
- Add courses, tracks, or majors (such as bioinformatics and gaming) that include emerging technologies.
- Develop concentrations or tracks in majors, where a portion of the upper-level and elective hours are used to emphasize high-level “Strategic Knowledge Areas” that build on standard CS or engineering curriculum.¹¹
- Provide programs that give in-depth emphasize on the area of computing or engineering practice in one discipline and provide a broad overview of other areas from the expanding knowledge areas.
- Introduce new majors to meet our changing educational needs.

George Mason University was among the first schools to recognize workforce trends and to initiate non-traditional undergraduate programs in response to the needs of the high technology community. Being an entrepreneurial and innovative University, The Volgenau School has answered this call by creating new majors in Information Technology (IT) and Applied Computer Science (ACS), and also creating new minors in computer science, information technology, data analysis, and software engineering. Mathematics is required by all these programs but plays a much more significant role in the areas of ACS and the traditional computer science and engineering majors than in the BS-IT major. As a mission-oriented School, our goal was to use the BS-IT major to help meet workforce needs by tapping into a broader pool of students who had “sufficient,” but not necessarily “strong” mathematical and programming aptitudes. The results would be an increased number of students drawn from a broader range of academic backgrounds and preparations pursuing technical degrees. Additionally, our aim was to promote the BS-IT major in a manner such that there would be an insignificant number of students attracted away from the traditional majors in computer science, computer engineering, and our other engineering disciplines. While current undergraduate academic programming appears to be meeting industry demands sufficiently, our Volgenau School of IT&E program managers can not be complacent; instead, they must communicate constantly with industry representatives to determine how to make their graduates even more competitive by implementing more appropriate and timely modifications to their curricula and program offerings, as well as developing effective strategies for informing high school counselors and prospective students and their parents.

In response to recent studies on enrollment trends, Blahnik² reports that St. Norbert College has significantly modified its computer science offering to include not only a traditional major in Computer Science but also to include concentrations in Business Information Systems and Graphic Design and Implementation. Cheng⁵ reports that the Rochester Institute of Technology (RIT) has, in addition to its traditional Computer Science major, quasi-specialized areas in Applied Software, Information Processing, Systems Software, and Computer Engineering. The Georgia Institute of Technology (Georgia Tech) is taking a much broader and drastic approach

by abolishing the core curriculum for computer science undergraduates.¹² In its place, Georgia Tech is introducing a curriculum called “Threads.” There are two main parts of the curriculum: in the first part, students pick two of eight possible threads of instruction; during their first year enrolled, students will also track themselves by “role,” picking among master practitioners (programmers), entrepreneurs, innovators or communicators. According to Jaschik¹², Richard A. DeMillo (dean of the College of Computing at The Georgia Institute of Technology) finds that “the one size fits all approach to computer science just isn’t working anymore,” and this is the underlying premise upon which Georgia Tech’s “threads” approach is based.

It is worth mentioning that the fields of computing and engineering are not alone in the problems that they face regarding the educational needs of its students. Jaschik¹³ also reports, for example, that there is a proposal for The Massachusetts Institute of Technology (MIT) to adopt major changes in undergraduate general science education. Some of the issues MIT is grappling with are of importance to many universities with strong science orientations. The issue at MIT has to do with moving from a highly prescribed curriculum to one with choice. Once again, it is acknowledged that the idea of a “core” in which everyone takes the same subjects may not be workable.

Responses to changes in technical workforce needs are not being addressed solely by engineering and computer science schools, especially when the problems are multi-faceted, and sometimes it is not easy to differentiate between the program offerings even when they reside within the same institution. As an example, the Applied Computer Science (ACS) major offered by The Volgenau School brings together computer science with disciplines such as biology and geography. For example, a strong background in computing may be used to analyze biological data that have been collected over several years. George Mason University’s newly formed College of Science is now proposing a new bachelor degree program in Computational and Data Sciences (CDS). This proposed degree in Computational and Data Sciences purports to represent a new direction for integrated science at George Mason University that is felt to be distinct from other existing undergraduate degrees, including our ACS degree. The rationale for creating this program is very similar to the rationale that was provided for creating the ACS program. Both programs claim that the public and private sectors need employees who can solve scientific problems with a computer, and these jobs require skills that are beyond either the individual traditional Computer Science or the applicable scientific discipline. Students in the ACS will have the opportunity to select computer science, and then they will choose a scientific or social science discipline area. On the other hand, students in the CDS curriculum will have the opportunity to select a specific scientific discipline, and then take computer science courses designed for specific scientific applications. Upon graduation, both ACS and CDS degree recipients will be productive (e.g., in the biological field) without requiring further training. It is predicted that there is a growing need for students produced by both the ACS and CDS majors, since biological data continue to be generated today, and it has been estimated that bioinformatics could be a \$2-billion industry within five years.³ Again, this is only one example of where the distinction between skills and knowledge possessed by graduates of differently named programs becomes blurred.

6. Viewing the Pool of Prospective Students Differently

In light of all of the changes that are taking place in the body of technical knowledge that need to be accommodated by educational programs provided by institutions, together with changing workforce practices and needs, a review of some of the “non-traditional” models employed by some of our educational institutions when determining entry and graduation requirements for the major is in order. According to Fernandez⁹, factors that must be taken into consideration when initiating curriculum redesign and development that meet today’s needs include, but are not limited to the following:

- Level of technical skills required in the major as well as in the profession
- Other specific technical skills needed
- Non-technical skills needed
- Amount of previous work experience required.

For an appropriately prepared computing professional, the list of technical skills may include programming languages, operating system and networking management, database developments and management, software engineering and project development. Other technical skills may include Web-based technologies, hardware installation, etc. The non-technical skills may include such things as interpersonal skills, leadership skills, communication skills, problem solving, self-motivation and initiative, accountability, and enthusiasm.

This model is in agreement with Cukier⁷, whose research suggests that mathematics is an indicator but not the only predictor of success in computing careers, and others who suggest that facility with languages is a better predictor of programming skills than mathematics intensive programs. It must be noted that traditionally, and even today, many who are responsible for designing computing curricula assume that soft skills (communication, project management, user needs assessment, etc.) are easily learned on the job. Non-traditional approaches to curriculum design such as, for example, selecting for soft skills and providing support or training for learning the other skills (e.g., on the job) are rarely brought into the discussion.⁷

Some departments have added the requirement that new programs must be flexible with regard to mathematics and science requirements. Others allow flexibility to the extent that the program meets the Accreditation Board for Engineering and Technology (ABET) requirements. As examples, designers of The Volgenau School of IT&E new information technology major have closely scrutinized the mathematics requirements that are essential to be effective for the IT professional, and have arrived at a collection of technical requirements that are far less cumbersome and unwieldy by comparison to technical requirements for engineering and computer science majors. All of this, however, was couched in the requirement that our information technology requirement meets evolving ABET requirements. On the other hand, The Volgenau School’s Applied Computer Science major, although including stringent mathematics and computing requirements, has given little consideration to ABET at any point in the design process.

7. Retaining Students in the Major

Getting properly prepared high school graduates to select computer science or engineering related majors is an important first step; however, much more must be done to retain them

through graduation. Once on campus, retention requires that we provide the following types of supporting and engaging activities:

- Exciting curricula related materials and activities
- Essential academic support services
- Opportunities for students to interact with faculty/researchers
- Work opportunities, field experiences, and other enriching educational experiences.

Innovative computing and engineering professionals, such as Denning⁸, who aim to slow and/or reverse the decline in the computer science discipline, have recommended adding an “innovation” theme to curriculum that would be very attractive to prospective students. Denning acknowledges that there are many challenges in such an approach. As examples, areas such as computer graphics, concepts from broadcasting, safety-critical or high integrity systems, multimedia, genomics, nana-materials, quantum computing, and tools such as Apple’s iMovies, provide interesting choices for the new innovative content. Core content would need to be repackaged and placed alongside “innovation material and practice” in the new courses. It is important to note that many faculty and staff would need to learn the practices of innovation themselves before teaching them.

Even when students enroll with a sound foundation in the courses essential for being successful in a computer science or engineering major, it is necessary to monitor the activities of these students to ensure that they remain directed and on tract. For students who enroll with weaknesses in their background, it is necessary to provide tutoring and other academic support services. It is equally important that the academically gifted students are challenged if they are to be retained in their chosen computer science or engineering major.

8. Impact of Non-traditional Programs on Traditional Programs

When the title for this paper was initially conceived, the research plan was a rather simplistic one that consisted of the following basic steps:

- Conducting a survey of universities that have started non-traditional bachelor degree programs/options to obtain enrollment trends data for traditional majors and the recently implemented non-traditional majors
- Analyze these data and report the findings.

Following the systems engineering approach that we strongly encourage for our computing and engineering students, the focus of the paper was broadened significantly after taking a closer look at “emerging trends” in the high technology environment as well as within the increasingly global economy. Admittedly, a paper entitled “Emerging Trends in Industry and Their Implications for Engineering Schools” is a better description for much of the content of this paper, and this content is representative of the “problem definition” phase of the systems engineering methodology.

Time, however, was devoted to examining the impact of non-traditional programs on traditional programs, and some of the findings are worth noting. For example, Cooke⁶ (University of South Carolina – Spartanburg) reports that the initial impact of their Information Management and

Systems (IMS) program on their computer science enrollments was dramatic. During the 2000-2001 academic year, approximately twenty percent of the computer science majors transferred to the IMS program. The sudden migration was a disturbing trend to the computer science faculty members. Since then the number of students exiting to IMS has declined to fourteen percent in 2001-2002 and twelve percent in 2002-2003. The concern among the computer science faculty members diminished as the IMS migration subsided and information became available as to why the students were leaving computer science. Interviews with computer science student transferring to IMS revealed that the primary reason for changing majors was a lack of interest in programming.⁶

Enrollment in our BS-IT program has exceeded original estimates and has soared to more than 900 majors this fall 2006 semester. Since the BS-IT program was implemented in 2002, enrollment in our Electrical Engineering, Computer Engineering, and Systems Engineering majors have remained stable, while enrollment in our Civil and Infrastructure Engineering program has increased significantly. In keeping with national trends, declining enrollment in our Computer Science major began prior to the introduction of the BS-IT major, and there is little evidence to suggest that this decline (which currently seems to be leveling off) is a result of the existence of our BS-IT program. Our BS-IT program has had its greatest impact on enrollments in the Decision Sciences and Management Information Systems (DMIS) program offered by our School of Management, where DMIS enrollment has plummeted from an enrollment of well over 600 students to less than 200 students since the introduction of our BS-IT program. Traditionally, many students enrolling in the DMIS major were “change of major” students from our computer science and engineering programs. We must keep in mind that “change of major” activities is not uncommon, since many students who initially express an interest in a computer science or engineering program do not understand what these majors entail, especially with respect to challenging technical courses. Every effort is made to retain students in the computer science and engineering majors; however, the BS-IT major and similarly designed majors provide an alternative that is well-suited to the academic preparation and needs of some of these students.

With time, the impact that non-traditional programs will have on traditional programs and academic units will be much clearer. At this time, however, the results have been mixed. Again, tracking emerging trends in industry and how educational institutions are responding to changing needs are far more valuable for computing and engineering program developer, managers, and administrators.

9. Continuing Challenges and Opportunities

A distinguished panel of the National Academy of Engineers (NAE)¹⁹ has written in *The Engineer of 2020* that “(engineering) students are still largely assigned to and educated in a single department, and as engineering disciplines have proliferated and clearly delineated specialties within these sub-disciplines have evolved, providing a broad engineering education to students has become an enormous challenge. ...” It is expected that as the nature of technical work continues to change during the 21st century, designers of academic programs will have to be more open and agile.

Many of the emerging engineering and computing disciplines require a multidisciplinary approach. The successful implementation of a major in an area such as bioengineering, for example, presents many challenges for which collaborating departments, which may be located within different units of the educational institution, may be unprepared. The interdisciplinary nature of bioengineering, e.g., requires close cooperation between engineers and biologists for it is clear that the development of quality curricula for bioengineering must draw upon the expertise of both disciplines.³

A more systematic strategic planning effort is required and should sensibly take into consideration how any new program will differ from other computing disciplines already present within a university. If this is not done, then confusion is likely to predominate. When talking with employers, high school counselors, and especially with prospective students and their parents, helping them understand how some of our majors differ in how they prepare students for careers is becoming increasingly challenging.¹ George Mason University's Applied Computer Science program and its Computational and Data Sciences program are prime examples of similarly designed programs that are not easily differentiated. Many employers are also frequently confused by the differences between some computer science programs and some information technology programs. These types of issues lead us to ask how do we ensure that universities are consistent in naming programs based on their content? If computing and engineering program designers increasingly disregard ABET, then there appears to be a need for some other institution and/or organizational structure for determining what degree should be awarded for a program's content in order to reduce the level of confusion among employers in the industry.¹

Another interesting question that administrators of innovative programs of a multi-disciplinary nature must address is related to staffing. The guidance provided by Yang¹⁸ could prove helpful. Yang suggests that one should begin by first examining the knowledge acquisition and transfer process in higher education institutions. Yang goes on to say that faculty play a central role in this acquisition and transfer process where an individual faculty member enhances his/her educational background and preparation by:

- Attending professional conferences and seminars
- Collaborating with major research institutions and centers
- Engaging in curricular revision activities
- Conducting theoretical/applied research
- Developing and offering topical workshops
- Interacting with students via classroom teaching and research activities.

The knowledge and experience acquired through these activities are then transferred to students via regular classes or workshops, as well as via publications. This acquire/transfer process is particularly characteristic of professionals in a computer science or engineering related field.

Many issues remain, and many of these issues can be indirectly attributed to successes of computing and engineering professionals. Examples of some of the remaining issues requiring attention are listed below, most of which are of a multidisciplinary nature and thus will require collaboration among departments and across larger academic units:

- Given the nature of existing educational institutions with their existing departments and program offerings, can they rise up to the challenge and meet current and emerging needs of the high technology community in the global economy, enabled by successes within the computing and engineering profession?
- How can we provide a greater stream of qualified and interested students so that one program or department is not competing with another for students in the pool?
- How do we ensure that advances such as on-chip massively-parallel processing, nano-systems and biologically-inspired computational systems are potential opportunities rather than threats to traditional engineering and computer science disciplines?
- Can a traditional area like computer science regain its standing as a discipline or will it necessarily continue to lose its popularity and break up into a number of application-specific academic programs?
- Will computer science as a major offering department survive rather than assume a “service-department” role, teaching primarily basic skill courses to incoming freshmen, technical courses to majors such as computer engineering, and/or technical courses integrated with existing majors such as biology, physics, geography, etc.?
- Would a refocusing of the computer science bachelor degree offering on areas such as computational biology, computational geography, gaming, embedded systems, and other novel aspects of computing lead to a more promising future for the computing professional?
- To what extent would an institution, especially one that is publicly funded, be allowed to redesign its curricula so that computing knowledge and skills that are critical to the business enterprise, and hence more difficult to successfully outsource, form the basis for the curricula and program offerings?
- How do employers ensure that students in traditional majors do not continue to receive priority in hiring, especially when their skills may not be as closely aligned with the job requirements as those of graduates from a non-traditional computing or engineering program?

This paper has identified and elaborated on several areas that should provoke thought within the community of scholars who will be responsible for developing and administering programs to meet the changing needs of technical communities during the 21st century. The issues are sufficiently complicatedly so that no single blueprint or model can be provided that will be well-suited to the unique needs of all institutions as they prepare to meet the needs of their individual constituencies. Again, instead of concentrating solely on impacts of today’s non-traditional computing and engineering programs on more traditional programs, the focus should be on emerging trends in the high technology environment as well as within the increasingly global economy. The progressive and entrepreneurial engineering schools that confront and embrace the identified and related challenges with an optimistic and “think out of the box” mindset will find unimaginable opportunities for new programming and organizational structures that will attract and retain both the “best and brightest” students and faculty. These will be among our most highly ranked engineering schools in both teaching and research for the 21st century.

10. References

1. Anthony, E. 2003. "Computing Education in Academia: Toward Differentiating the Disciplines." In *Proceedings of the 4th Conference on information Technology Curriculum* (Lafayette, Indiana, USA, October 16 - 18, 2003). CITC4 '03. ACM Press, New York, NY, 1-8.
2. Blahnik, J., McVey, B., and Pankratz, D. 2006. "Adding Concentrations To the CS Major: Our Dean Calls Us 'Innovative.'" In *Proceedings of the 37th SIGCSE Technical Symposium on Computer Science Education* (Houston, Texas, USA, March 03 - 05, 2006). SIGCSE '06. ACM Press, New York, NY, 191-194.
3. Burhans, D. T. and Skuse, G. R. 2004. "The Role of Computer Science in Undergraduate Bioinformatics Education." In *Proceedings of the 35th SIGCSE Technical Symposium on Computer Science Education* (Norfolk, Virginia, USA, March 03 - 07, 2004). SIGCSE '04. ACM Press, New York, NY, 417-421.
4. Cardoso, João M. "New Challenges in Computer Science Education," in 10th ACM Annual Conference on Innovation and Technology in Computer Science Education (ITiCSE'05), Universidade Nova de Lisboa, Lisbon, Portugal, June 27-29, 2005, ACM Press.
5. Cheng, R. 1976. "A Diversified Undergraduate Computer Science Program." In *Proceedings of the ACM SIGCSE-SIGCUE Technical Symposium on Computer Science and Education* (February 01 - 01, 1976). R. Colman and P. Lorton, Eds. ACM Press, New York, NY, 366-370.
6. Cooke, D. W. 2003. "A Multidisciplinary Information Management and Systems Program: Pearl or Peril?" In *Proceedings of the 4th Conference on information Technology Curriculum* (Lafayette, Indiana, USA, October 16 - 18, 2003). CITC4 '03. ACM Press, New York, NY, 13-22.
7. Cukier, W., Shortt, D., and Devine, I. 2002. "Gender and Information Technology: Implications of Definitions." *SIGCSE Bull.* 34, 4 (Dec. 2002), 142-148.
8. Denning, Peter J. and Andrew McGettrick. "Recentring Computer Science." *Communications of the ACM* 48.11 (2005): 15-19.
9. Fernandez, John D. and Phyllis Tedford. "Evaluating Computer Education Programs Against Real World Needs," *The Journal of Computing Sciences in Colleges*, vol. 21, no. 4, April 2006, 259-265.
10. Geiger, Gordon. "The Engineering Management Curriculum at the University Of Arizona: A Template for Undergraduate Engineering Education." *A New Model for Undergraduate Engineering Education*. Tuscan: American Society for Engineering Education, 2006.
11. Hoganson, K. 2005. "A Strategic Approach to Computer Science Curriculum." In *Proceedings of the 43rd Annual Southeast Regional Conference - Volume 1* (Kennesaw, Georgia, March 18 - 20, 2005). ACM-SE 43. ACM Press, New York, NY, 365-370.
12. Jaschik, Scott, "New 'Threads' for Computer Science," *Inside Higher Ed* (insidehighered.com), Sept. 26, 2006.
13. Jaschik, Scott, "New 'Threads' for Computer Science," *Inside Higher Ed* (insidehighered.com), Oct. 16, 2006.
14. Kershenbaum, A., Hadimioglu, H., Ivanov, L., Schiaffino, R., and Hoffman, M. 2006. "Is Computer Science Still Relevant?" *J. Comput. Small Coll.* 21, 6 (Jun. 2006), 182-187.
15. Kumar, A. N., Beidler, J., Bhagyavati, Farian, H., Haas, M., Kushleyeva, Y., Lee, F., and Russell, I. 2005. "Innovation in Undergraduate Computer Science Education." *J. Comput. Small Coll.* 21, 2 (Dec. 2005), 138-142.
16. Saad, A. and Boisvert, D. 2005. "National Science Foundation Advanced Technological Education Projects and Centers of Excellence for Information Technology Education: An Overview." In *Proceedings of the 6th Conference on information Technology Education* (Newark, NJ, USA, October 20 - 22, 2005). SIGITE '05. ACM Press, New York, NY, 233-238.
17. Stiller, E., and LeBlanc, C. (2003). "Creating New Computer Science Curricula for the New Millennium." *Journal of Computing in Small Colleges*, 18(5), 198-209.
18. Yang, Andrew. "Computer Security and Impact on Computer Science Education". *The Journal of Computing in Small Colleges*, Proceedings of the sixth annual CCSC northeastern conference on The journal of computing in small colleges, Volume 16 Issue 4. April 2001.
19. *The Engineer of 2020*, National Academy of Engineering, National Academics Press, Washington, D.C., 2004.