A Comparison of Baccalaureate Programs in Information Technology with Baccalaureate Programs in Computer Science and Information Systems

Han Reichgelt
Georgia Southern University,
Statesboro, GA, USA

han@georgiasouthern.edu

Barry Lunt
Brigham Young University
Provo, UT, USA

luntb@byu.edu

Tina Ashford
Macon State College
Macon, GA. USA

tashford@mail.maconstate.edu

Andy Phelps
Rochester Institute of Technology
Rochester, NY, USA

amp5315@it.rit.edu

Erick Slazinski
Purdue University
West Lafayette, IN, USA

edslazinski@tech.purdue.edu

Cheryl Willis
University of Houston
Houston, TX, USA

clwillis@uh.edu

Executive Summary

A number of universities have recently started to add baccalaureate programs in Information Technology (IT) to their existing programs in Computer Science (CS) and (Management) Information Systems (IS). While some have welcomed this development, others have argued that (a) there are significant differences between the baccalaureate programs in IT that different institutions currently offer, and that it is, therefore, impossible to speak of a generic IT baccalaureate degree, and that (b) to the extent that there is a similarity, there is no real difference between programs in IT, and programs in CS or IS.

There are two ways in which one can try to settle the question whether there are significant differences between IT programs and programs in CS or IS. One method is what one might call top-down. Such an approach essentially involves arriving at a generally agreed-on definition of the subject, or at a consensus on the capabilities that graduates in the discipline must have, and then using these definitions to determine whether there are significant differences.

A top-down methodology has three drawbacks. First, it does not reflect the way in which degree programs in new areas tend to emerge. Programs in new areas are typically designed by individual institutions to meet some perceived demand, and they are not designed based on a general definition of the area that a relevant community has agreed on beforehand. Second, a top-down comparison obviously does not allow one to answer the question whether there are significant similarities between IT programs currently being offered by different institutions. Finally, the definition that one is likely to accept for a particular discipline depends on one's perception of the fields under consid-

eration. Thus, if one believes that there is a significant difference between, say, IT and IS, then one is likely to prefer definitions of these fields that emphasize these differences. On the other hand, if one is not convinced that there is a difference, then one is more likely to accept definitions that minimize the difference.

These considerations have led us to adopt a bottom-up approach to the question whether there are significant similarities between IT programs offered by different institutions and whether they differ sufficiently from programs in CS and IS. We surveyed the programs in the areas of IT, IS and CS being offered at 10 different institutions across the United States, and classified the courses that students were required to take as belonging to one of seven categories, namely Business; Electronics and Signals; Hardware; Interpersonal Communication; Networking, Web Technologies and Databases; Mathematics, Physics and Chemistry; and, finally, Software. While we found some differences between the IT programs offered at the different institutions, the similarities between them seemed more significant. Moreover, we found that there were significant differences between programs in IT, programs in IS and programs in CS.

Introduction

Recently, a number of universities in the United States and elsewhere have started baccalaureate programs in Information Technology (IT). In addition to the universities with which the authors are affiliated, these institutions include Capella University, Illinois State University, Indiana University, Pennsylvania College of Technology, the State University of New York at Morrisville, the University of Baltimore, and the University of South Alabama. Most of the institutions in question have added the baccalaureate degree in Information Technology to other computing-related baccalaureate degrees, such as degrees in Computer and/or Electronic Engineering, Computer Science, Information Systems, Computer Information Systems, and Management Information Systems.

While some (e.g., Denning, 2001) welcome this development, others are less accommodating and argue that there is nothing that would make a baccalaureate program in IT sufficiently distinct from a baccalaureate program in an existing computing discipline to warrant a separate degree program. The argument that IT baccalaureate programs are not sufficiently distinct is most often heard from faculty in programs in Computer Science (CS) and (Management) Information Systems (IS). The argument is often two-fold. First, it is argued that there are significant differences between IT programs and that one, therefore, cannot speak of a single (type of) IT baccalaureate degree. Second, to the extent that there is any similarity between IT degrees offered at different institutions, there is no significant difference between them and degrees in CS or IS.

There is some anecdotal difference that there is confusion on the part of prospective students and their parents concerning the difference between degrees in CS, IS and IT. Academic advisors in the institutions offering all three types of degree report spending a good deal of their time explaining the differences between the programs to incoming students. This confusion is exacerbated by the fact that the term "Information Technology" is also used as an umbrella term to encompass CS, IS and IT in the narrower sense that we are using the term in this paper.

There are two methodologies that one can use to try to shed light on the question whether there are any major differences between CS, IT and IS. The first can best be described as **top-down** and involves first arriving at an agreed-on definition of the various disciplines involved, and then comparing the definitions to see whether there are important differences between the disciplines. A second methodology can best be described as **bottom-up**. It involves an empirical analysis of the structure of the actual degree programs that various universities offer at this point in time. In particular, this analysis asks, "Are there noticeable similarities in the structure of the actual IT

programs that different institutions currently offer and, if so, are there major differences between those programs and other computing-related degree programs?"

Top-down Methodology: Defining a Discipline

The top-down methodology can be conducted in at least two different ways, following different ways of defining a discipline. The first and most traditional way of defining a discipline is to come up with a one or two sentence description of a discipline.

For example, CS can be defined as follows:

> Computer Science is the study of the design and properties of algorithms, and their linguistic and mechanical realization

while IS could be defined as:

> Information Systems as a field of academic study encompasses the concepts, principles and processes for two broad areas of activity within organizations: (1) acquisition, deployment, and management of information technology resources and services (the information systems function) and (2) development, operation, and evaluation of infrastructure and systems for use in organizations processes (system development, system operation, and system maintenance). (Gorgone, Davis, Valacich, Topi, Feinstein, and Longenecker; 2002, p. 11.)

Finally, IT could be defined as follows:

> As an academic discipline, Information Technology focuses on meeting the needs of users in an organizational and societal context through the selection, creation, application, integration and administration of computing technologies.

An alternative way of defining a discipline is by the skills that one expects professionals in the discipline to possess. This way of defining a discipline fits in well with the recent emphasis on outcomes-based accreditation criteria that has been adopted by accreditation agencies such as ABET in the United States. Moreover, there are various documents that are generally accepted by the relevant communities as defining program outcomes for different disciplines. For example, the Joint Task Force on Computing Curricula of the IEEE Computer Society and the Association for Computing Machinery lists the skills in Table 1 as required for all computer science graduates (Joint Task Force, 2001, p. 64).

Table 1: Capabilities and skills for computer science graduates.
Source: Joint Task Force, 2001, p. 64
Cognitive capabilities and skills relating to computer science • Knowledge and understanding. Demonstrate knowledge and understanding of essential facts, concepts, principles, and theories relating to computer science and software applications. • Modeling. Use such knowledge and understanding in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoff involved in design choices. • Requirements. Identify and analyze criteria and specifications appropriate to specific problems, and plan strategies for their solution. • Critical evaluation and testing. Analyze the extent to which a computer-based system meets the criteria defined for its current use and future development. • Methods and tools. Deploy appropriate theory, practices, and tools for the specification, design, implementation, and evaluation of computer-based systems.

- Professional responsibility. Recognize and be guided by the social, professional and ethical issues involved in the user of computer technology.

Practical capabilities and skills relating to computer science
- Design and implementation. Specify, design and implement computer-based systems.
- Evaluation. Evaluate systems in terms of general quality attributes and possible tradeoffs presented within the given problem.
- Information management. Apply the principles of effective information management, information organization and information-retrieval skills to information of various kinds, including text, images, sound and video.
- Human-computer interaction. Apply the principles of human-computer interaction to the evaluation and construction of a wide range of materials including user interfaces, web pages and multimedia systems.
- Risk assessment. Identify any risks or safety aspects that may be involved in the operation of computing equipment within a given context.
- Tools. Deploy effectively the tools used for the construction and documentation of software, with particular emphasis on understanding the whole process involved in using computers to solve practical problems.
- Operation. Operate computing equipment and software systems effectively.

Additional transferable skills
- Communication. Make succinct presentations to a range of audiences about technical problems and their solutions.
- Teamwork. Be able to work effectively as a member of a development team.
- Numeracy. Understand and explain the quantitative dimensions of a problem.
- Self-management. Manage one's own learning and development, including time management and organizational skills.
- Professional development. Keep abreast of current developments in the discipline to continue one's own professional development.

Gorgone et al. (2002, p. 6) give the skills in Table 2 as those required for an Information Systems graduate.

Table 2: Capabilities and skills for information systems graduates
Source: Gorgone et al., 2002, p. 6

1. IS professionals must have a broad business and real world perspective. Students must therefore understand that:
   - IS are enablers of successful performance in organizations
   - IS span and integrate all organizational levels and business functions
   - IS are increasingly of strategic significance because of the scope of the organizational systems involved and the roles systems play in enabling organizational strategy.
2. IS professionals must have strong analytical and critical thinking skills. Students must therefore:
   - Be problem solvers and critical thinkers
   - User systems concepts for understanding and framing problems
   - Be capable of applying both traditional and new concepts and skills
   - Understand that a system consists of people, procedures, hardware, software and data.
3. IS professionals must exhibit strong ethical principles and have good interpersonal communication and team skills. Students must understand that:
   - IS require the application of professional codes of conduct
   - IS require collaborations as well as successful individual effort
   - IS design and management demand excellent communication skills (oral, written, and listening)

- IS require persistence, curiosity, creativity, risk taking and a tolerance of these abilities in others
4. IS professionals must design and implement information technology solutions that enhance organizational performance. Students must therefore:
   - Possess skills in understanding and modeling organizations processes and data, defining and implementing technical and process solutions, managing projects, and integrating systems
   - Be fluent in techniques for acquiring, converting, transmitting, and storing data and information
   - Focus in the application of information technology in helping individuals, groups and organizations achieve their goals.

Finally, the Society for Information Technology Education, in a draft of accreditation criteria for baccalaureate programs in Information Technology, lists the skills required for an IT graduate shown in Table 3.

Table 3: Skills and Capabilities for Information Technology Graduates
Source: Society for Information Technology Education, 2003.

1. Use and apply current technical concepts and practices in the core information technologies, including human computer interaction (HCI), programming, computer networking and hardware, databases, web technologies and issues;
2. Analyze, identify and define the requirements that must be satisfied to address problems or opportunities faced by organizations or individuals;
3. Effectively design IT-based solutions and integrate them into the user environment taking into account user-centric design and interface, usability testing for effectiveness and efficiency, and the impact of the solution on the wider organization;
4. Create an effective project plan, including accurate estimates of the time, financial and other resources required for completion of the project, and implement the plan, including the timely identification and implementation of appropriate measures to stay within schedule and budget constraints;
5. Identify and evaluate current and emerging technologies and discuss their applicability to solve the users' needs;
6. Analyze the impact of technology on individuals, organizations and society, including ethical, legal policy issues;
7. Demonstrate independent critical thinking and problem solving skills;
8. Collaborate in teams to accomplish a common goal by integrating personal initiative and group cooperation;
9. Communicate effectively and efficiently with clients and peers both orally and in writing, using appropriate terminology;
10. Recognize the need for continued learning throughout their career.

Inspecting the skills that these sources expect graduates to have, one notices a number of similarities. For example, as one would expect of any baccalaureate degree program, all stress an importance of analytical and critical thinking skills, ethical issues, communication skills and the need for life-long learning. Moreover, all stress the importance of being able to identify and analyze user requirements.

On the other hand, the draft criteria also reveal a number of differences. For example, only the CS criteria mention the importance of the theoretical underpinnings of computer science and software development, while only the IT draft criteria mention the importance of project management skills.

A top-down comparison of different computing programs throws some interesting light on the question whether there are significant differences between CS, IT and IS. However, the methodology is limited. First, it ignores the way in which new programs typically emerge. New programs in the professions tend to emerge relatively spontaneously in response to some perceived need. It is rare that a set of like-minded people first agrees on a definition or a set of program outcomes before they start offering programs in that particular discipline. Generally accepted definitions for a discipline or standards for a particular discipline, whether they are accreditation standards or curriculum standards, are typically formulated only after a number of programs have emerged in that discipline. For example, it is not a coincidence that CS was the first discipline to have generally accepted accreditation standards and a model curriculum, with IS following. On the other hand, for IT the accreditation standards have not been formally approved yet and there is, as yet, not even a draft of a model curriculum. The availability of such material reflects the relative maturity of the fields.

Second, while a top-down comparison may help one decide whether there are significant differences between CS, IT and IS, it answers only one of the two questions that we set out to answer in this paper, namely whether there are significant similarities between the IT programs that different institutions currently offer.

Finally, a closer inspection of both the generally accepted definitions of the three disciplines under consideration, or of the program outcomes in Tables 1 through 3 reveals that there is a great deal of interpretation possible. Therefore, a comparison of programs based on these program outcomes is likely to rely too much on an individual's interpretation of the program outcomes since no measurement standards are specified. One could argue that the information in Tables 2 and 3, for example, can equally well be used to support the position that there are significant differences between programs in information systems and information technology, as to support the position that they are the same. Which interpretation one prefers is likely to be colored by one's prior views on the difference between CS, IS and IT. In other words, the definition of the field one prefers, whether it is in the form of a general statement or in the form of a set of skills that graduates in the field will acquire, is likely to be heavily influenced by one's perceptions of the fields in the first place. Therefore, we argue that, in the absence of a clear, unambiguous and generally agreed on definitions of the fields under consideration, an exercise that attempts to identify differences between them is likely to be fruitless.

It is for these reasons that in this paper we approach the question of whether there are significant differences between different programs in computing in a bottom-up fashion. We inspected the computing-related degree programs in 10 different institutions to determine whether (a) there are significant similarities between the different programs in IT currently being offered at the various institutions around the country, and whether (b) there were significant differences between IT programs and programs in CS and IS.

The organization of the paper then is as follows. In section 2, we discuss the methodology of the study in more detail. Section 3 reports the raw results of the study, while in section 4 we analyze the data. Section 5 is the conclusion.

Methodology

In determining the structure of the various computing programs, we classify the courses that make up the degree program into one of 7 categories. The categories are as follows:

- Business related courses (B)
- Courses concentrating on interpersonal communication (IC)
- Software related courses (SW)

- Courses on networking, web-related technologies or databases (NWD)
- Electronics and signals (ES)
- Hardware (HW)
- Mathematics and Science (MS)

We ignored general education courses taken by all students at a particular institution because they are not unique to a given major and, therefore, do not help define differences between majors. It is important to stress that the mathematics and science requirements included in our analysis are courses required over and above the mathematics and science requirements that form part of general education requirements in most US universities.

The reason for classifying courses in this way was not arbitrary. In December of 2001, about 40 representatives from 15 schools currently offering baccalaureate programs in IT attended the first Conference on Information Technology Curricula (CITC I) in Aspen Grove, Utah, to discuss a number of IT-related issues. One of the topics for discussion was IT curricula and delegates at the conference engaged in an exercise to capture their views on what a core IT curriculum should contain. Each delegate was asked to list on a separate piece of paper each topic that they felt was important in an IT curriculum. Some 700 pieces of paper were collected, obviously with some duplication of topics.

Participants were then asked to classify the topics thus collected into a number of categories, and their frequencies noted. 28 categories emerged from this exercise, of which the most frequent were the following:

- Networking
- Interpersonal communication
- Software
- Web systems design
- Databases
- Business related issues (e.g., project management, e-business, organizational structure)
- Digital communications
- Data security/privacy
- Mathematics
- Systems design
- Hardware

Further details can be found in Lunt, Helps, Lawson, and Goodman (2002).

These topics were then used as a starting point in a pilot study to analyze the various computing and related degrees offered at Brigham Young University (BYU), Utah. In addition to Information Technology, BYU offers baccalaureate degrees in Computer Engineering, Computer Science, Electrical Engineering, and Management Information Systems. It was on the basis of this pilot study that several of the categories were combined to arrive at the seven categories mentioned earlier and used throughout this study.

In order to simplify understanding the comparison, we used the equivalent of three-semester credit hours as our base unit. (Semester hours are widely used in the United States, and increasingly elsewhere, as a measure of course load. 1 semester hour is equivalent to 1 lecture hour per week over a 15 week period, or 3 supervised lab hours per week over a 15 week period.) Thus, if a course was listed as 2 semester credit hours, it was given a number of .667.

One further issue needed to be resolved. Many programs commonly allow students options in coursework (called variously as minors, specializations, or such) and the question of how to count these options arose. The methodology that we used was simply to score each course by a ratio.

Thus, if the student was required to choose two out of five 3-credit courses to satisfy an option, then each course was given a rating of .4 (2/5). For example, the B.S. in Computer Science at the University of Houston requires a 15-hour minor in either Science/Systems or Business. Each business course within the option therefore gets a rating of .5, and the option as a whole a rating of 2.5.

In addition to his or her own institution, each author also analyzed the programs offered at one other institution. The only requirement was that the institution in question offered a baccalaureate program in IT. This led to programs at the following institutions being analyzed:

- Brigham Young University, Provo (BYU)
- Capella University (Capella)
- George Mason University (GMU)
- Georgia Southern University (GSU)
- Macon State College (MSC)
- Pennsylvania College of Technology (PCT)
- Purdue University (Purdue)
- Rochester Institute of Technology (RIT)
- University of Baltimore (UB)
- University of Houston (UH)
- University of South Alabama (USAl)

Some institutions offered multiple programs in Information Technology, as opposed to options within a general IT degree program. For example, Purdue University offers four different degree programs with different titles in its Information Technology program. Such programs were rated separately because the program name appears on the graduates' diplomas. On the other hand, if a school offered options within a single IT program, then such programs were simply rated once. For example, Georgia Southern University offers four specializations within each general BS program in Information Technology, but since these are specializations within a single IT program, we arrived at a single rating for its IT program. The basis for the decision on whether to treat programs as separate or as specializations within the same general program was made based on the name of the degree program that appeared on graduates' diplomas.

The selection of universities to include in this study was unsystematic. However, the requirement that the university offer a degree in IT, as well as degrees in CS and IS, severely limited the population of possible candidates. As stated, IT has only recently emerged as a separate discipline and there therefore are not many universities in the US currently offering degree programs in IT. However, the universities included in the study provide a good cross-section of the different types of universities one finds in the US. Note that, unlike the academic institutions elsewhere, universities in the US are often classified according to the Carnegie Classification of Institutions of Higher Education (www.carnegiefoundation.org/Classification). The Carnegie classification distinguishes between Baccalaureate Universities and Colleges, institutions that award baccalaureate degrees only, Master's College and Universities, institutions that, in addition to baccalaureate degrees, also offer Master's degrees, and Doctoral Granting institutions, institutions that also award doctoral degrees. Our selection included representatives from each class. Moreover, one of the institutions included in our sample, namely Capella University, offers its degrees exclusively through distance education.

Results

Table 4 reports our findings for selected Computer Science programs.

Table 4: Breakdown of baccalaureate programs in Computer Science by topic area.								
The numbers represent the quantity of 3-semester-credit-hour courses required by each program in each topic area.								
	B	**ES**	**HW**	**IC**	**NWD**	**MS**	**SW**	**Total**
BYU-Utah		2.5	3	2	1.82	6	13	28.32
George Mason Univ		2	4		5	2	5	18
GSU		1	2	1	2.5	8	9.6	24.1
Purdue		2			2	11	16.7	31.67
RIT - CS	1.2	0.67	2.81	3	5	12.2	4	28.85
RIT - Software Engineering	1	1.11	2.81	1	2	11.8	5	24.75
Univ of Houston	2.5					8.5	14	25
Univ of South Alabama			1	2	2	5	13	23
Average (n=8)	.59	1.16	1.95	1.13	2.54	8.06	10.03	25.46

Table 5 gives the breakdown of the number of 3-semester hour courses in selected IT programs by topic area.

Table 5: Breakdown of Information Technology programs by topic area.								
The numbers represent the quantity of 3-semester-credit-hour courses required by each program in each topic area.								
	B	**ES**	**HW**	**IC**	**NWD**	**MS**	**SW**	**Total**
BYU-Utah	1	3.9	3.1	2	4	5	3	22
Capella	11.37			1	16	0	6.75	35.12
George Mason Univ	5		1	1	10	1	1	19
GSU	2		0.25		7	3.19	4.63	17.07[1]
MSC	4		2	2	11		5	24
MSC: New Media	3		2	2	6		2	15
MSC: Systems Analysis	9		2	2	8		4	25
PCT: Data Comm & Networking	3	2	1.58	4	8.25	4	7.25	30.08
PCT: Internetworking Appl Dev.	3		1.58	4	4.25	4	12.25	29.08
Purdue: Databases	6			6.33	8.5	4	5.5	30.33
Purdue: Networking	3	4	1	4.33	10.5	5.67	4.5	33
Purdue: Systems Analysis	6			6.33	5.5	4	7.5	29.33
Purdue: Software Development	6			5.33	4.5	4	9.5	29.33
RIT: Information Technology	3		1	3.72	13.7	6	3	30.44
RIT: New Media	3		1	3	7	6	4	24
Univ of Baltimore: Applied IT				2	6.67	3	5.33	17
Univ of Houston (IST)	6			2	9	2	4	25
Univ of South Alabama				2	15	3	5	25
Average (n=18)	4.13	.55	.92	2.96	8.60	3.16	5.23	26.04

Table 6, finally, gives our data for programs in (Management) Information Systems.

Table 6: Breakdown of (Management) Information Systems programs by topic area. The numbers represent the quantity of 3-semester-credit-hour courses required by each program in each topic area.								
	B	**ES**	**HW**	**IC**	**NWD**	**MS**	**SW**	**Total**
BYU-Utah	15			2	3		3	23
Capella: E-business	31			1				32
GSU	13			1	4.8	1	4.2	24
PCT	18.67		0.33	3.5	2.17	2	3.33	30
RIT	9		1	2	5	3	3	23
Univ of Baltimore	10			2.1	5.2	2	5.5	24.8
Univ of Houston	12				8	3		23
Univ of South Alabama	6			1	6	1	2	16
Average (n=8)	14.33	0.00	0.17	1.58	4.27	1.50	2.63	24.48

Analysis

We set out to answer two questions in this paper, namely whether there are significant similarities between baccalaureate programs in IT offered at different institutions, and whether there are significant differences between IT programs and programs in CS or IS

We tackle the second question first. Figure 1 shows the average percentage of courses in each topic area for the different discipline areas. It readily shows where the emphasis lies for programs in IS: the vast majority of the required courses are in Business (58%). CS programs, on the other hand, emphasize Software (39%) and Math, Physics, & Chemistry (30%), while programs in IT put most emphasize on networking, web technologies and databases (33%).
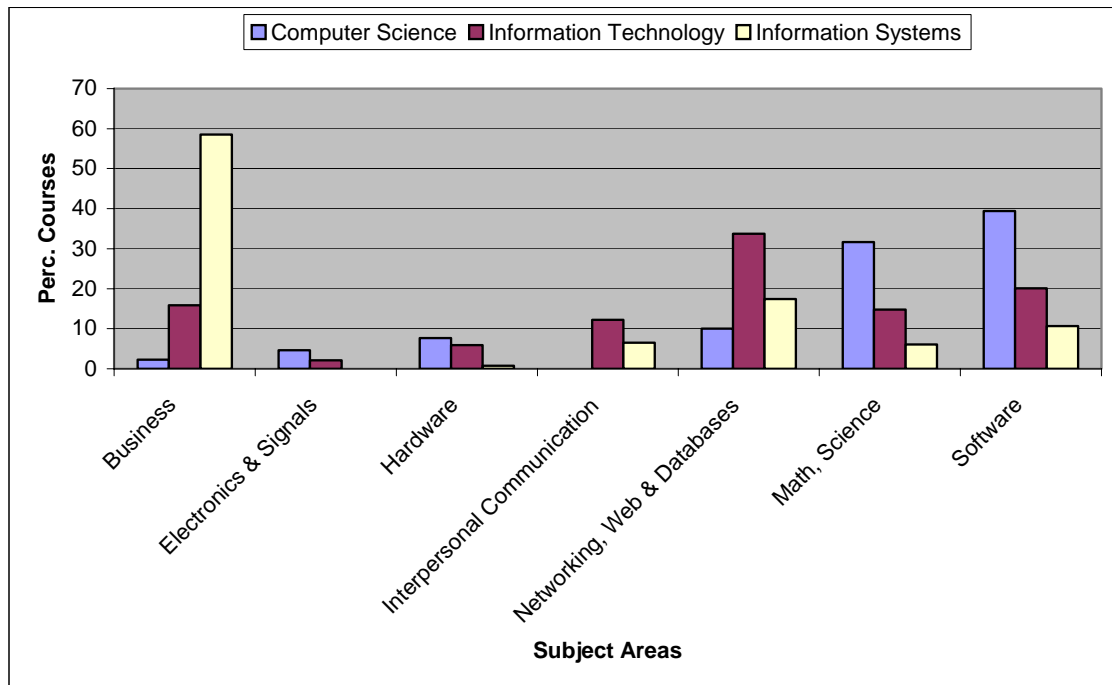

Figure 1: Breakdown of courses in each area for computing programs

Table 7 summarizes the data on the distribution of courses between the "average" programs in CS, IS and IT. A χ^2 analysis reveals that the differences between the average programs are highly significant ($\chi^2 = 33.91$, df = 12, p = .0007)

Table 7: Distribution of courses between average programs in CS, IS and IT								
	B	**ES**	**HW**	**IC**	**NWD**	**MS**	**SW**	**Total**
Average CS program	.59	1.16	1.95	1.13	2.54	8.06	10.03	25.46
Average IS program	14.33		.17	1.58	4.27	1.50	2.63	24.48
Average IT Program	4.13	.55	.92	2.96	8.60	3.16	5.23	25.55
Total	19.05	1.71	3.04	5.67	15.41	12.72	17.89	

While we see noticeable differences between the average programs in CS, IT and IS, the question remains whether there are important similarities between the programs. We tackled this question in two ways. First, we performed a χ^2 analysis on the distribution of the courses in the different categories for each of the programs (the data were presented in tables 4-6). The χ^2 values that we obtained were 55.51 for IS (df = 42), which is not significant at any level; 64.09 for CS (df = 42), significant at p = .015), and 156.41 for IT (df = 102), significant at p = .0004. In other words, there were no significant differences between programs in IS, significant differences for programs in CS, and highly significant differences for programs in IT.

The explanation of the greater uniformity within IS programs may lie in the nature of any available model curricula (recall that there are currently no model IT curricula). The CS model curriculum consists of a relatively large number of knowledge units, combinations of skills and concepts that can typically be covered in a few contact hours. It encourages institutions to package the knowledge units into courses in any way it sees fit (although it includes examples of how this can be done). By contrast, the IS model curriculum provides guidance at a much higher level. It consists of a set of specific 3-semester hour courses. The reason for the greater prescriptiveness of the IS model curriculum may lie in the fact that most IS programs are offered in business schools that also seek accreditation by the AACSB and the AACSB accreditation standards, until recently, were themselves rather prescriptive. In other words, the greater uniformity of the IS programs may be the consequence of the availability of a more specific IS model curriculum.

However, it is also possible for the relatively high χ^2 value for CS and IT programs to be caused by a few outliers. We therefore performed a second analysis to determine if there were such outliers. For each of the programs, we determined the distance to the average CS, IS or IT program. Since not every program required the same number of courses, we first determined the percentage of courses in each category both for the "average programs" and for the various programs in our study. We then calculated the difference for each program to the average programs by adding the absolute differences between the percentages of courses offered in each category. Tables 8-10 give the actual calculations for the different programs in our study.

Table 8: Distances of actual CS programs to average CS, IS and IT programs			
	Dist. to Average CS Program	Dist. to Average IS Program	Dist. to Average IT Program
BYU-Utah	3.27	13.91	9.58
George Mason Univ	7.78	13.00	6.98
GSU	.60	13.58	9.38
Purdue	3.62	15.37	11.75
RIT - CS	5.56	10.91	7.24
RIT - Software Engineering	4.30	13.24	9.09
Univ of Houston	5.33	14.63	11.43
Univ of South Alabama	4.28	13.46	9.23
Average CS Program		13.14	8.93

Table 9: Distance of actual IT programs to average CS, IS and IT programs			
	Dist. to Average CS Program	Dist. to Average IS Program	Dist. to Average IT Program
BYU-Utah	6.94	10.80	7.28
Capella	13.13	7.32	5.62
George Mason Univ	13.50	7.95	6.16
GSU	8.09	10.66	4.06
MSC	10.96	9.60	3.55
MSC: New Media	12.46	8.93	4.33
MSC: Systems Analysis	11.92	5.73	4.85
PCT: Data Comm & Networking	7.21	9.71	2.48
PCT: Internetworking Appl Dev.	4.94	10.21	5.41
Purdue: Databases	10.39	7.89	2.74
Purdue: Networking	8.97	9.89	3.26
Purdue: Systems Analysis	8.81	7.76	4.13
Purdue: Software Development	7.45	8.18	4.81
RIT: Information Technology	10.08	9.91	3.87
RIT: New Media	7.49	9.21	2.82
Univ of Baltimore: Applied IT	7.32	11.85	4.38
Univ of Houston (IST)	11.05	6.25	2.84
Univ of South Alabama	10.72	11.85	5.27
Average IT Program	8.93	8.47	

Table 10: Distance of actual IS programs to average CS, IS and IT programs			
	Dist. to Average CS Program	Dist. to Average IS Program	Dist. to Average IT Program
BYU-Utah	14.04	2.24	9.81
Capella: E-business	18.91	7.67	16.14
GSU	12.37	1.86	7.60
PCT	13.43	2.04	9.23
RIT	10.57	3.88	4.88
Univ of Baltimore	10.61	3.78	5.17
Univ of Houston	14.93	4.85	7.56
Average IS Program	13.14		8.47

Figures 2-4 represent the data graphically. Figure 2 shows the distances of the actual program to the average CS and IS program; Figure 3 the distances to the average CS and IT programs, and Figure 4 the distances to the average IS and IT programs.

When we inspect both the data and scatter diagrams, a few things become apparent. First, as one would expect based on the χ^2 analysis of the data, the programs in IS in general seem to be more tightly clustered than the programs in either CS or IT. However, there is one glaring exception, namely the e-business program offered at Capella University. Second, in general, IT programs seem to lie somewhere between IS and CS programs. This is perhaps shown more clearly in figure 2, where we see a cluster of IS programs in the upper left hand corner, a cluster of CS programs in the lower right hand corner, with the cluster of IT programs somewhere in between. Finally, even though there is considerable variation in the programs in IT, with the exception of the IT program at BYU Utah, all the programs in IT are closer to the average IT program than they are to the average CS or IS program. In other words, although there is significant variation in the IT programs (as shown by the χ^2 analysis), it still appears to be the case that, with the exception

of the program at BYU Utah, all IT programs are more similar to the average IT program than they are to either the average CS or IS program.
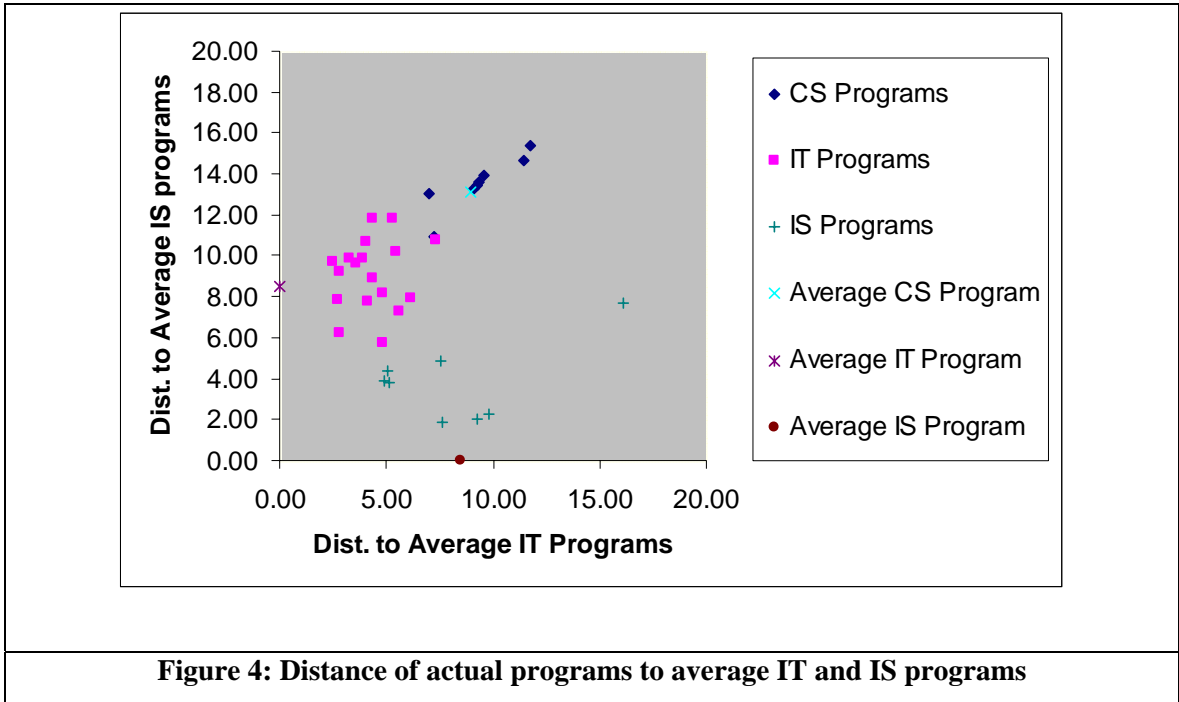

Figure 2: Distance between actual programs and the average CS and IS programs

Figure 3: Distance of actual programs to average CS and IT programs
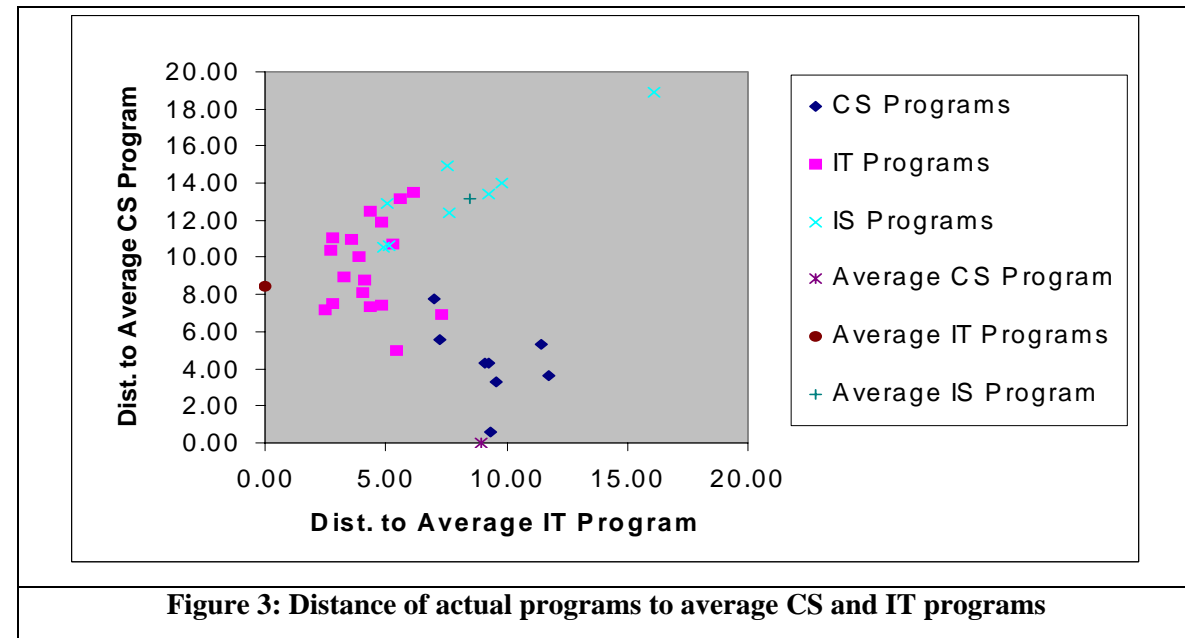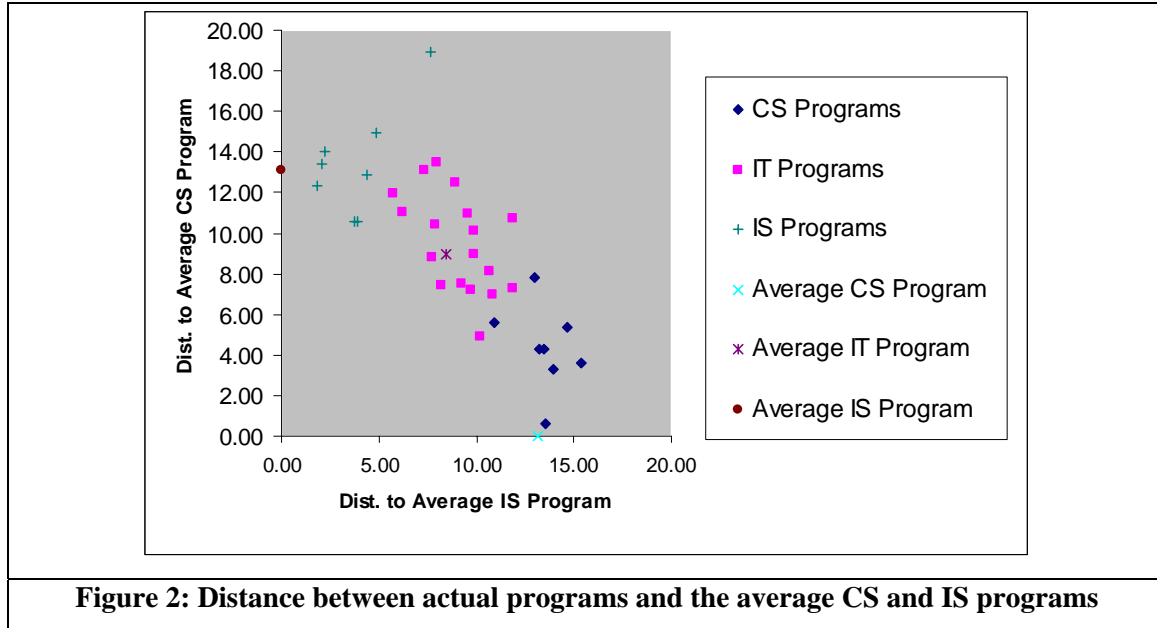
Figure 4: Distance of actual programs to average IT and IS programs

Conclusion

Our empirical comparison has shown a distinct difference in profile between baccalaureate programs in Computer Science, Information Technology and (Management) Information Systems. It has also demonstrated that there is a greater uniformity in programs in (Management) Information Systems than there is in programs in Computer Science programs or in Information Technology. Clearly, the fact that Information Technology programs have not been in existence for a long period, and the current lack of generally agreed accreditation and/or curriculum standards, makes the latter hardly surprising. Nevertheless, despite the lack of such material, Information Technology programs seem still more similar to the average Information Technology program than they are to either the average Computer Science or (Management) Information Systems program.

Obviously, our analysis is crucially dependent on the validity of our classification scheme. One could for example argue that the category of courses in networking, web technologies and databases is too broad. In a similar fashion, our category of business courses contains both what one might describe as business content courses (e.g., courses in Economics, Finance or Management), as well as courses in project management. While the latter are typically taught in business schools, there is an argument that such courses instill in students skills and knowledge that are distinctly different from the skills and knowledge that students acquire as a result of business content courses. This point may be particularly pertinent to our analysis of the difference between Information Technology and (Management) Information Systems programs as the largest difference between actual Information Technology programs seems to lie in the amount of courses in the business category that students are required to take (figures range from none to more than 11).

Nevertheless, we believe that our data suggest that the answer to both questions that we started out to answer, namely are there important similarities between different programs in Information Technology, and are they sufficiently different from programs in either Computer Science or (Management) Information, Systems is affirmative. In our view, the data give sufficient evidence for the proposition that actual IT programs are more similar to each other than they are to programs in Computer Science or (Management) Information Systems.

References

Denning, P. (2001). The IT schools movement. *Communications of the ACM, 44* (8), 19-22.

Gorgone, J., Davis, G., Valacich, J., Topi, H., Feinstein, D. & Longenecker, H. (2002). *IS 2002: Model curriculum and guidelines for undergraduate programs in information systems.* Available from http://www.acm.org/education/is2002.pdf

Joint Task Force on Computing Curricula. (2001). *Computing curricula 2001-Computer science.* Available from http://www.computer.org/education/cc2001/final/index.htm

Lunt, B., Helps, R., Lawson, E. & Goodman, G. (2002). Designing an IT Curriculum: The results of the first CITC conference. *Proceedings of the 2002 American Society for Engineering Education Annual Conference & Exposition.*

Reichgelt, H., Zhang, A. & Price, B. (2002). Designing an information technology curriculum: The Georgia Southern University experience. *Journal of Information Technology Education*, *1* (4), 213-222. Available online at http://www.jite.org/documents/Vol1/v1n4p213-221.pdf

Biographies

Han Reichgelt holds degrees in Philosophy and Psychology from the University of Nijmegen, Netherlands, and a PhD in Cognitive Science from the University of Edinburgh, Scotland. He joined Georgia Southern University in 2001 from the University of the West Indies, Mona, Jamaica. Over the last few years, he has been active in efforts to create accreditation standards for baccalaureate programs in IT. He is the (co-) author of some sixty journals and refereed conference papers, as well as of a textbook on Knowledge Representation in Artificial intelligence. His research interests include IT pedagogy and the use of IT in low-income countries.

Barry Lunt is associate professor of IT at Brigham Young University in Provo, UT. He has been in the teaching profession for 17 years, all at the post-secondary level. He has varied professional experience, including 7 years as a design engineer with IBM, 6 summers as a faculty intern, and several involvements in consulting. His academic degrees include a BS and MS in Electronics Engineering Technology from BYU-Provo, and a PhD in Occupational and Adult Education from Utah State University. His publications are in the areas of electronic physical design and engineering education. At BYU, Barry has heavily involved in the transition from an EET program to an IT program, and was one of the founders of the present CITC conferences. He remains actively involved in defining IT on a national basis.

Tina Ashford is an Assistant Professor at Macon State College in Macon, Georgia. Her primary areas of responsibility include curriculum development of introductory, hardware, and history of Information Technology courses. Ms. Ashford actively presents at regional and national conferences, and has been an invited guest lecturer at numerous local civic meetings. Her tech-help column, "The TechnoFiles", appears weekly in the Houston Home Journal, the legal organ of Houston County, Georgia. She is the recipient of many awards, with her latest being selected for the 10th edition of 2000 Notable American Women.

Andrew Phelps is an assistant professor at the Rochester Institute of Technology, in Rochester, NY. He is the founding faculty member of the Game Programming Concentration within the Department of Information Technology and his work in games programming education has been featured in The New York Times, CNN.com, USA Today, National Public Radio, IEEE Computer and several other articles and periodicals. He regularly publishes work exploring game engines and their education uses, both academically and at the Macromedia DevNet Center, and is currently at work on his first text with Prentice Hall. He maintains a website at http://andysgi.rit.edu/ featuring his work as an educator, artist, programmer, and game addict, and currently teaches courses in multimedia programming, game engine development, 2D and 3D graphics, and information technology theory.

Erick Slazinski is an Assistant Professor of Computer Technology at Purdue University in West Lafayette. He holds an M.S. in Computer Science from the Florida Institute of Technology. His interests include business intelligence, data analysis, database design, data warehousing, database engine technologies and knowledge management.

Cheryl Willis is an Associate Professor of Information Systems Technology at the University of Houston. She received her Ph.D. in Curriculum and Instruction from the University of Florida. Her teaching focus is primarily on applications development and database management. Her current research interests include service learning in information technology undergraduate programs and the use of emerging technologies in undergraduate teaching. She has developed curriculum for business education and information technology at the secondary, post-secondary, undergraduate, and graduate levels. She is involved this year in studying the use of tablet PCs in a m(obile) Learning Lab.