

AC 2007-930: ON A COMBINED ELECTRICAL ENGINEERING AND COMPUTER SCIENCE DEPARTMENT

Dick Blandford, University of Evansville

Dr. Dick Blandford is the department chair of the Electrical Engineering and Computer Science Department at the University of Evansville. He holds a PhD from the University of Illinois.

Deborah Hwang, University of Evansville

Dr. Deborah Hwang is the program director of the Computer Science program at the University of Evansville. She holds a PhD in Computer Science from MIT.

On a Combined Electrical Engineering and Computer Science Department

Abstract

In the United States, more than 10% of traditional electrical engineering programs have combined with computer science into a single department¹. Historically computer science programs emerged from mathematics or electrical engineering departments in the mid to late 1960's. Since mathematics is not typically in the engineering college many computer science programs developed independently from engineering programs. In the 1970's and early 1980's, with the advent of the microprocessor and embedded computer systems, computer engineering evolved mostly out of existing electrical engineering programs and has since become a separate discipline distinct from electrical engineering with its own accreditation process. It is common to see electrical engineering and computer engineering in the same department.

Computer engineering course work complements courses in computer science and in many cases there is considerable duplication. Thus, it makes sense to consider a combined electrical engineering and computer science department. At the authors' university, electrical engineering and computer science have been combined into a single department since 1990, with computer science, computer engineering, and the more traditional electrical engineering as ABET accredited programs within that department.

A combined department offers the obvious advantages of administrative efficiency and elimination of duplication. There are many not-so-obvious advantages as well in the areas of program assessment, shared resources, team projects, and the new ideas and opportunities that are generated when programs are mixed. Disadvantages of a combined department have arisen as well.

This paper presents the advantages and disadvantages of a combined department, details the methods used to accomplish the successful combination, and provides advice for those contemplating such a change.

Introduction

Electrical engineering and computer engineering programs commonly reside in a single department with a single department chair and separate program directors. They have common core and elective coursework, and often share assessment objectives, outcomes, and rubrics. In recent years a few universities, particularly smaller universities, have combined a computer science program with computer engineering and electrical engineering into a single department with separate program directors. Beginning in the year 2000 when computer science accreditation came under the ABET umbrella, this alternative has become more attractive and easier to accomplish.

The computer science program at the University of Evansville began in 1965 as a program out of the mathematics department in the College of Arts and Science. In 1980, the program was

moved from Arts and Science to the College of Engineering which is now called the College of Engineering and Computer Science but the computer science program retained its separate department status. In 1990, mostly for reasons of administrative efficiency, computer science was combined with computer engineering and electrical engineering into a single department. Electrical engineering was first ABET accredited about 35 years ago and computer engineering was first ABET accredited in 1996, due in no small part to the support and cooperation of the computer science program. In 2005 computer science became an ABET accredited program and this too can be attributed, at least in part, to the successful combined department.

This paper outlines the workings of a combined department and gives details of the administrative efficiencies as well as faculty members' perspectives on the combination. The overall experience has been successful and generally positive but also has had a few disadvantages.

History

Computer science at the University of Evansville began fittingly as a subprogram of the mathematics department. A capricious administrator moved computer science to the College of Engineering in 1980. In the College of Engineering, now called the College of Engineering and Computer Science (CECS), computer science remained a separate department along with departments of electrical, mechanical, and civil engineering. Computer engineering became a program within the electrical engineering department about this same time. Due to a decline in computer science enrollment in the late eighties, and in the name of administrative efficiency, electrical engineering, computer engineering, and computer science were combined into a single department with a single department chair and three program-director positions.

As a result of this change in organization, the computer science program changed focus. The parts of the program which dealt with business systems were dropped. Computer science students were required to take the same math and science sequence as the electrical and computer engineering students. Electives in the computer science program emphasized topics such as artificial intelligence, graphics, and networks, and deemphasized databases and file structures.

The computer engineering program became ABET-accredited in 1996 due in no small part to the combined department and the change in emphasis in computer science on more engineering related topics. Computer science at that time had no real plans for accreditation mostly because the Computer Science Accreditation Board (CSAB) required a minimum of five faculty in a computer science program and accreditation seemed unnecessary as computer science graduates were able to find jobs regardless of program accreditation.

In the year 2000, when CSAB joined with ABET and the Computing Accreditation Commission (CAC) was created, there was a sense that accreditation in computer science would become much more important to industry, graduate schools, and prospective students. In 2006 the computer science program at the University of Evansville was accredited for the first time. This too was attributed in large part to the combined department and the robust atmosphere that it provided.

The Mechanics of the Switch

The primary reason for creating a combined department was administrative efficiency. A secondary reason was the ongoing need for quality computer science support for the computer engineering program. Having computer science and computer engineering in the same department has made for better cooperation between the two programs in the way of research, course offerings, and capital resources.

Throughout the 1980's there had been ongoing and probably natural tension between the needs of the computer engineering program and the computer science program. (At one point the computer engineering program had their own version of a computer architecture class.) But by 1990 the large enrollments of the computer science program had declined dramatically due at least in part to a nationwide down cycle. Simultaneously, the computer science department chair left the university for a different job and several computer science faculty were nearing retirement. After the departments were combined, a program director in computer science was hired to replace the former department chair. The computer science program director refocused the computer science program, changed much of the curriculum, and convinced the remaining computer science faculty that the changes were for the better.

Computer Science Curriculum Changes

The effect of being in a combined department is pervasive, starting in the first year. In most computer science programs, Java has been the choice for the first language. Students coming out of high school with Advanced Placement (AP) credit have taken a test in Java to demonstrate a working knowledge of the language. While computer engineering has incorporated Java in many places, most text books have examples in C++, and C remains the dominant language for embedded applications. Since both computer engineering and computer science students share the first programming courses, we chose to retain C++ as the first language and teach Java in a later course required only of computer science students. If the programs were not combined into a single department that shared some courses, it is likely that the first programming courses in computer science would use Java.

Another major change in the computer science curriculum consists of the way in which the technical electives are structured and chosen. Upper level electives in the computer science program are structured such that they are open to both computer science students and to computer engineering students. This means that in general, the only prerequisite for these courses is the data structures course. In terms of which courses are offered, for example, it means that an advanced version of object oriented programming will probably not be offered unless there is great enrollment in computer science since there is no room in the computer engineering curriculum for the necessary prerequisites. Likewise, computational linguistics, advanced databases, distributed operating systems, or web design are not likely to be offered as computer science electives.

In general computer engineering students view topics in a more pragmatic manner than do computer science students as documented by student comments at both the course and

curriculum levels. Examples where this has had an effect include the computer science course in programming languages, which focuses more on language implementation than it does on language semantics. Another example is the networks course that has a significant component in information theory and hardware, including a lab, and does not focus exclusively on programming, network topologies and transmission protocols.

Electrical and Computer Engineering Curriculum Changes

Computer engineering is a relatively new area of engineering and there exists a wide spectrum of computer engineering programs ranging from those which are, in effect, specialized electrical engineering programs, to those which have a decidedly strong software component. In a combined department computer engineering tilts toward the software side. As a result, the computer engineering program attempts to graduate those who are as competent in software as they are in hardware.

Toward this end, the junior year of the computer engineering curriculum has three courses which enhance the computer engineering student's programming abilities. (1) The first course in microcontrollers requires students to mix assembly language and C in a term long project. (2) A course on small systems software, which formerly focused on device drivers and interfacing software, has become more of a systems programming course that includes network programming and high level programming of hand held devices. (3) Finally, the real-time programming class builds on the programming skills from the computer science data structures class and specifically aims to prepare students for the computer science operating systems class. As a result of these three courses, computer engineering students emerge from the junior year with considerable software skills and are better prepared to take on the more advanced computer science courses, both required and elective.

The natural part of computer science that can be taught by the electrical and computer engineering faculty is that component related to hardware. The computer architecture course, for example, is taught by a computer engineering faculty member. The first course in microcontrollers, primarily a computer engineering course, has a heavy software component and does not require a course in circuits as a prerequisite. It is open to computer science majors who have had the machine organization class, which is also taught by an electrical or computer engineering faculty member. The course in real time programming, taught by the computer engineering faculty, is an upper-level required course for computer engineering students, but is open to computer science students as well. Such courses are designed with the idea that they may be taken by either computer science students or computer engineering students.

In the electrical engineering curriculum, only a single all-software course is required. This course parallels that taken by the computer engineering and computer science students but has less emphasis on program design and documentation. However, electrical engineering students do acquire a firm foundation in object-oriented programming. In the junior year they take a course that requires a working knowledge of C as well as assembly language for two different processors.

Over the last five years, as a result of assessment data, the faculty have examined the programming content of the electrical engineering program, and at this time we are still contemplating additions to the curriculum. In a survey completed online in 2003, we found that very few electrical engineering programs required more than a single programming course and many of those programs concentrated on C or even MATLAB with no object-oriented programming. However, a small but significant percentage of our electrical engineering graduates have jobs in which they are required to use programming; this group has expressed the opinion that a second programming class should be added to the electrical engineering curriculum. It seems likely that over the next two years we will make this change.

Assessment

Since computer science, computer engineering, and electrical engineering are all ABET accredited programs and since the assessment process can be grueling and tedious, we have made an effort to get as much efficiency in assessment from our combined department as possible. Thus we have a common assessment rubric and we have, after much debate, a common set of objectives for the three programs with different outcomes.

The rubric for all three programs makes use of the "assessment day" concept². One day per semester we meet as a department for the express purpose of assessing all of the course work for that term. Faculty establish course objectives and outcomes for each course and every course is examined by a small group of faculty who have not taught the course in order to determine whether the outcomes have been met and to make suggestions for changes. This means that a faculty member in computer science may be part of an assessment team for the sophomore circuits course, or conversely, an electrical engineering faculty member whose specialty is power systems may be on an assessment team to examine a course in operating systems.

The ABET documentation for assessment is greatly reduced since a common rubric is used. Courses which are taken by both computer engineering majors and computer science majors can be done with a single assessment rather than two assessments with different rubrics.

The assessment process also has become a uniting factor that all three programs have in common. Faculty in one program not only gain a better understanding of their own curriculum as a result of assessment; they also understand and support what is going on in other programs as well since they are part of the process of assessment.

Faculty Perspective

From a faculty point of view there are three areas of impact for a combined department: crossover of course content, crossover of effective teaching methods, and wider use of resources.

Faculty in a combined department necessarily develop a broader perspective with regard to the courses that they teach. For example, the "hard-core" electrical engineering faculty in our program understand what is in the computer science course in data structures since they have participated in the assessment process of that course. In some cases, we have had faculty in one area "sit in" on courses in another area because of this crossover interest. Because of intense

graduate school specialization, computer science faculty necessarily have a different perspective on solving problems than does, for example, an electrical engineer who has specialized in control systems. The combined department provides a fertile atmosphere for crossover which would be otherwise difficult to obtain. One good example of this occurs in the freshman intro course in which computer science, computer engineering, and electrical engineering students take a common course which introduces all three disciplines. The course focuses on robotics and at one time made extensive use of LEGO™ robots^{3,4} as a vehicle to introduce both hardware, software, and the combination of the two. Likewise, at the senior year it is common for faculty in one discipline to sponsor projects involving a student or a student team with members of another discipline. The combined department effectively removes bureaucratic barriers, and work among the three disciplines is limited only by faculty's willingness to extend themselves outside their traditional areas.

Because the working group of colleagues is more diverse in a combined department, ideas that prove to be successful tend to spread rapidly within the department. (We define the working group of colleagues as that small group of people who regularly exchange professional ideas. This may include methods of teaching as well as ideas about technical matters, creativity, and invention.) For example, the electrical engineering faculty began teaching the sophomore circuits sequence as an integrated lab/lecture course. That class meets two hours a day three days a week for a combined lab/lecture session. The professor may lecture for 20 minutes and follow up immediately with a one hour lab followed by more lecture⁵. This idea was carried over to the first course in programming taught by the computer science faculty. This programming course is taken by both computer science and computer engineering students and now meets two days a week for two hours a day. The professor again mixes lab with lecture on an as-needed basis. Similarly, the senior project course in computer science formerly had more procedural structure than the senior project course in electrical engineering. Electrical engineering faculty then "borrowed" several of the ideas and concepts used in the administration of the computer science senior project course. This included formal mid-term progress reports and assignment of grades by faculty committee rather than by individual faculty. These are two examples of effective crossover of teaching methods and procedures among the three disciplines.

With regard to new faculty in any of the three areas, we look for faculty with a particular area of interest as well as an ability to work in at least one of the two other areas outside of her own program. For example, we were fortunate to hire a person in the electronics area who also had a working knowledge of real time Linux systems. In addition to a course or two in electronics, he also teaches the course in real-time programming and a Unix systems programming elective.

Also, in our combined department, the labs of the three disciplines are viewed by faculty and students as belonging not to a single program, but to the department. The computer science program has a teaching lab and a student project lab, along with the appropriate equipment to support those labs. Computer and electrical engineering have an array of laboratories including a networks lab and labs for electronics, optics, control systems, circuits, and projects. It is common to see computer science students working in electrical engineering project labs, and electrical engineering students using the resources in the computer science project lab. From the viewpoint of both faculty and students, we appear to have more space and resources than we would have if we were in separate departments.

The Upside

There are genuine advantages to the combined department arrangement. Each advantage along with a short explanation is listed below:

- **Robust programs:** The computer science program becomes more robust because there is a wider audience for electives. Computer engineering students tend to be academically strong as a group and incorporating these academically able students into a computer science course creates competition and a new perspective. Also computer engineering becomes more robust because computer science courses are designed and structured with computer engineering students in mind.
- **Elective choices:** Computer engineering has a crowded curriculum without much room for electives in our University's version of the curriculum. The computer science program, on the other hand, has several electives and many more choices. Since we have combined the programs, and we create computer science electives with computer engineering in mind, the computer engineering students benefit from a wider array of choices.
- **Course content:** With computer engineering and computer science in the same department, and a shared assessment process, there is a finer level of control over course content. Changes in course content in one program can be immediately reflected in changes in another program without passing through the usual departmental (or college) bureaucracy.
- **Collaboration:** Freshman intro courses and the senior project are obvious places where the three programs can collaborate. Our freshman intro course is an introduction to all three majors and at the senior project level we often have faculty and student crossover on a single project. The common assessment process encourages this. We have had some collaboration on research, conference papers, and projects but the level of collaboration has not met our expectations.
- **Administrative efficiency:** This comes in three forms: (1) there is more flexibility in scheduling of classes, class rooms, and faculty; (2) the crossover in faculty mobility between programs is common and would otherwise be rare; (3) there are more resources for each program in that computer science has access to hardware that they might not otherwise have and computer and electrical engineering have access to computer labs and servers that would otherwise be off limits. The programs also share support personnel and have a common budget.

The Downside

All is not rosy and positive. There are disagreements and arguments and the "us" and "them" mentality often threatens. Here is a list of the disadvantages of a combined department:

- **Common ground:** Computer science faculty and electrical engineering faculty develop by means of a different educational process. As a result the two groups tend to think differently about course content, teaching methodology, and what is important in an

undergraduate education. This results in a less coherent group of faculty which can be more difficult to lead.

- Curriculum constraints: Both the computer science curriculum and the computer engineering curriculum become a bit more constrained. To achieve some level of efficiency in the combined department it is essential that the two programs mesh well. Each cannot go its own way without regard for the other's students.
- Unconvinced faculty: Even after having a combined department for 15 years there are still faculty who would like to see the department split. This comes in part, from the natural friction of working together and from the tendency of humans, and maybe faculty in particular, to see greener grass in other pastures. Without continued support from the dean, department chair, and the program directors the department might split.
- Size: When the EE and CS departments combine to form a large department there is a point where the merger no longer makes sense. Our feeling is that the merger makes most sense for departments in which fewer than 20 faculty are involved.

Future Directions

We have considered and are indeed still considering, making the first three semesters of computer engineering, computer science, and electrical engineering identical. This would mean that the computer science students take the first circuits course along with the computer and electrical engineering students. It also means that the electrical engineers take the data structures and algorithms programming course currently taken by the computer science and computer engineering students.

Advice, Counsel, and Comment

It takes about five years for the "us" and "them" ideas to dissipate. The majority of our faculty (12 in all) support the combined department. Such a change cannot be done without the enthusiastic support of the dean, the department chair, and the program directors. Like the assessment process, the change needs a champion or better, several champions who are enthusiastic supporters.

It is critical that once the change is made that the three programs begin to do as many things in common as is practicable. Common department meetings are essential. If the computer science faculty meets separately from the electrical engineering and computer engineering faculty, differences between the two groups tend to become entrenched and difficult to resolve. We also believe having a common assessment process is vital.

ABET appears to be supportive of this arrangement and recognizes the advantages, particularly for small programs, on both the computer science and the engineering side. It is clear that the combined department played a significant role in the accreditation of both computer engineering and computer science.

Hiring new faculty who support the combined department appears to be an easier alternative than convincing older faculty that the change is desirable. This is natural and to be expected since faculty who are in place already have publicly stated opinions which become difficult to retract.

However, in our opinion, faculty who are in place and who become convinced that a combined department is a positive development tend to be more supportive and add stability to the change. It is worth the effort it takes to convince in-place faculty that the combined department is in their interest.

Bibliography

1. Batarseh, Issa (moderator), "EE, CompE and CS Programs: Merger or Peaceful Co-Existence?", ECEDHA '06 discussion session, Oahu, Hawaii..
2. Blandford, D. and Hwang, D., "Five Easy but Effective Assessment Methods" 2003 ACM SIGCSE Technical Symposium and Workshops, Reno, Nevada, February 19-23, 2003.
3. M. Goldweber, C. Congdon, B. Fagin, D. Hwang, F. Klassner, "The Use of Robots in the Undergraduate Curriculum: Experience Reports" panel presented at 2001 ACM SIGCSE Technical Symposium and Workshops, Charlotte, NC, February 21-25, 2001.
4. D. Hwang & D. Blandford, "A Multidisciplinary Team Project for EE's CoE's, and CS's" ASEE Annual Conference, St. Louis, MO, June, 2000.
5. Reising, James, "Sophomore Circuits Course Sequence Revision, An Integrated Laboratory/Lecture Approach", ASEE Annual Conference, St. Louis, MO, June, 2000.