

## Are We Scientists or Engineers?

W. A. WULF

*University of Virginia, Charlottesville, Virginia <wulf@virginia.edu>*

Like those sophomoric debates about trees falling in the forest, the answer depends on the definition.

In my dictionary, the preferred definition of science involves discovering properties of the natural world. Engineering involves the application of those properties to solve practical problems. By those definitions we are neither scientists nor engineers. The principal objects of our study are manmade, not natural.

The third definition in my dictionary is more expansive, focuses more on the method of discovery, and includes the rigorous study of any phenomena. By that definition we are perhaps both scientists and engineers, since we both study and apply computation. But I am not sure that is a useful response to the original question, since it reflects only that the level of abstraction has been raised—at a some even higher level of abstraction we are probably also humanists. If we abstract too much, we lose the intent of making the distinction in the first place.

Does the answer matter? In some ways, I suppose so. It's handy to have an analogy to help explain what we do. Certainly we share a lot with both science and engineering—the scientific method on the one hand and the construction of useful, reliable, and cost-effective artifacts on the other. No one can contest that the foundations of computation are as deep and intellectually rich as those of any natural science, and the systems we engineer are clearly among the most sophisticated ever devised. Since most people have positive feelings about both science and engineering, the analogies are generally useful, even if they aren't perfect and even if the listener doesn't really understand what it means to be a scientist or engineer anyway.

The problem with analogies, however, is that they cut both ways. If we ever decided the question, we would have to live with inferences that we might not like. As a case in point, there is a movement in engineering education to structure the core curriculum on the common applications of the conservation laws and the second law of thermodynamics. Proponents of this view recommend that, to be accredited, the first two years of all engineering curricula be organized exclusively around these. If we're engineers, then presumably that applies to us.

I, for one, believe that computing professionals need the earliest possible introduction of discrete mathematics and formal methods. My own depart-

---

Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the ACM copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Association for Computing Machinery. To copy otherwise, or to republish, requires a fee and/or specific permission.

© 1995 ACM 0360-0300/95/0300-0055 \$03.50

ment happens to be in an engineering college, and I am delighted that our students get a solid grounding in the physical sciences, continuous mathematics, and engineering design—but it would be profoundly wrong to preclude discrete mathematics from the beginning of their curriculum. Computing professionals need the mathematical tools to allow them to reason about computational structures, and they need to have those tools exercised throughout their education; that can't happen if discrete mathematics is delayed into the third year. Because of limited space, I'll leave as an exercise to the reader listing some of the really unpleasant consequences of being pegged as scientists.

Beyond the danger of others drawing improper inferences, letting ourselves be captured by the analogy is even more dangerous. If we are one then we are not the other; and so those others do not belong in our academic departments or professional societies. Defining ourselves by what we are not is common in CS and it has, in my opinion, damaged us. It has made us inhospitable to pragmatically and intellectually fertile areas that should be part of our discipline. Indeed, beyond being inhospitable, we have expelled whole areas—numerical methods, libraries, and MIS, for example. If we continue this expulsion of the practical we will leave the field a barren husk.

The tendency to define ourselves by what we are not is, I think, a legacy of the early days of the field when many were concerned about being seen as merely programmers for the *real scientists*. We were not just programmers. Sadly, echoes of this outdated fear are present in the title question itself. Some people seem to think that one answer is more respectable. In truth, we will not be more respected because of what we call ourselves.

Let's remember that there is only one nature—the division into science and engineering, and subdivision into physics, chemistry, civil and electrical, is a human imposition, not a natural one. Indeed, the division is a human failure; it reflects our limited capacity to comprehend the whole. That failure impedes our progress; it builds walls just where the most interesting nuggets of knowledge may lie. The fact that other disciplines have chosen to build those walls is a failure that we are not required to emulate.

Young as we are, I think we don't really have a choice to be science or engineering; we are science and engineering, and something more too. Evidence abounds for the value of the interplay in our field between the science and engineering, the theory and practice, the basic and applied. The most theoretical aspects of complexity theory are motivated by the most pragmatic of concerns, performance. New results in number theory and logic are both driven by and drive practical issues of security and privacy in networked systems. The practical application of formal methods would have sounded like an oxymoron to mathematicians less than a century ago, but now is a goal within sight.

The fact is, whatever you call us, CS is the place to be. Our young and slightly paranoid discipline is in the vanguard of changes in science, technology, and society that are as profound as any precipitated by the traditional sciences and engineering—and probably more so.

It might be more comfortable to define ourselves by analogy to the establishment, but what we do is too important to be limited by that. Let's kick the inferiority complex and find our own way, define our own paradigms. Let's

adopt the applicable best from science, engineering, the social sciences, and the humanities. Let's be inclusive rather than exclusive. Let's build even more bridges to other disciplines and to industry; let's amplify the special strengths of each. Like the immigrants to the U.S., industrial affiliations and applications in diverse disciplines enrich and vitalize our core.

I reject the title question. I call my discipline Computer Science because that's the common name in the U.S.—but my discipline spans a multidimensional spectrum from deep and elegant mathematics to crafty programming, from abstraction to solder joints, from deep truth to elusive human factors, from scholars motivated purely by the desire for knowledge to practitioners making my everyday life better. It embraces the ethos of the scholar as well as that of the professional. To answer the question would be to exclude some portion of this spectrum, and I would be poorer for that.