

CIPS Position Paper

Software Engineering

written by Paul G. Bassett M.Sc., I.S.P.

(October 2004)

Introduction

Software engineering is a relatively young discipline. Recently the engineering community has sought to restrict who has the right to teach it, and even more importantly, has intimated their intention to restrict who has the right to practice what the engineering profession deems to be software engineering. The university and computer science (CS) communities are disputing such restrictions. What follows is intended to provide members of CIPS and the public with an understanding of this ongoing dispute, and to express the CIPS position with regard to the issues involved.

Background

CIPS is heavily involved in both the academic and professional aspects of IT. In 1982 CIPS established the Computer Science Accreditation Council (CSAC). This autonomous council accredits four-year university programs in computer science and software engineering. CIPS also established autonomous councils to accredit management information systems programs and college diploma programs. Graduation from an accredited program reduces the years of experience a CIPS member needs to become certified as an Information System Professional of Canada (I.S.P.).

In 1997, the Canadian Council of Professional Engineers (CCPE) and The Association of Professional Engineers and Geoscientists of Newfoundland sued the Memorial University of Newfoundland (MUN). They alleged MUN's use of "software engineering" (SE) in the name of its SE program infringed their trademarks because it was delivered by the computer science department rather than engineering. The Canadian Engineering Accreditation Board (CEAB, analogous to our CSAC) wants to accredit all engineering programs - only graduates of CEAB-accredited programs are eligible for "P.Eng." licenses. (One is supposed to be able to take an exam in lieu of being such a graduate, but few if any provincial engineering bodies have bothered to set one.)

The CCPE recommends each provincial engineering Act contain this definition: —The "practice of professional engineering means any act of planning, designing, composing, evaluating, advising, reporting, directing or supervising, or managing any of the foregoing, that requires the application of engineering principles, and that concerns the safeguarding of life, health, property, economic interests, the public welfare or the environment." The CCPE also recommends this exemption (... denote **extraneous words**): —Nothing ... shall prevent an individual who ... holds a recognized honours degree or higher in one or more of the ... computer ... sciences ... from practicing ... any act (including management) requiring the application of scientific principles, competently performed." CCPE does not define the meanings of —engineering principles" and —scientific principles", rendering the definitions sufficiently circular that provincial and territorial associations may deem a fairly arbitrary class of activities to be professional engineering. (Similar Acts prevent unqualified people from practicing in other professions -- what the College of Physicians and Surgeons deems to be "practicing medicine", or what the Canadian Bar Association deems to be "practicing law", to name but two.) In various jurisdictions engineering associations are changing their Acts in a manner that appears to make it easier to further restrict the title and practice of software engineering.

On September 21, 1999 the lawsuit against Memorial University was halted with a 10point agreement that included a five-year moratorium from litigating the use of CCPE's official marks - "engineer" and "engineering" - and the establishment of the Software Engineering Panel (SWEP). The latter was to deliberate and make recommendations regarding the use of the term "software engineering" in the undergraduate university community. The SWEP was comprised of an impartial chair and one representative from each of science academia, engineering academia, the engineering profession, and the IT profession (Paul Bassett represented CIPS). The Panel, operating in a spirit of compromise, solicited input from all interested parties. In July 2000, the Panel issued a unanimous report.

Right on the heels of the SWEP came its successor, the "SE Task Force" (Paul Bassett again represented CIPS), mandated to translate the recommendations into specific proposals for accomplishing joint CSAC/CEAB accreditation. Given there was little or no effort to get the various stakeholder groups to "buy into" the SWEP's report, no one should have been surprised when the Task Force proposals hit an impasse. As a result, both CSAC and CEAB independently accredit SE programs mounted within their separate constituencies, and the issues behind the lawsuit remain unresolved.

The moratorium mentioned above expires in July 2005. In addition to issues already cited, two key threats have emerged that continue to challenge the non-engineering (i.e., non P.Eng.) computer science community. First, the —right to title“ issue has not been resolved so it is unclear if this community can continue to use terms like —Software Engineer“ following the moratorium or if it will once again be challenged. (The key issue here is the need to understand that Software engineering's roots and core concepts are clearly in Computing and Information Technology, and not the other way around. If Software Engineering exists in some intersection between Computing and Engineering then people from both areas should have the right to such titles.) Secondly, and no less important, is the —right to practice“ autonomously. To date, there has been no clear statement from the engineering community that they respect our right to practice. (This could be done by including exclusionary clauses in the various acts in the country, which is not without precedent.) The implications of both of these issues clearly has important consequences to the practicing computer professional and the students currently enrolling in computer science programs in the country.

Discussion

A fundamental question underlies this dispute: who has the right to practice software engineering? The Engineering profession is aiming - through existing engineering statutes -to make the practice of "software engineering" illegal by anyone who lacks a "P. Eng." license. This would affect not just those calling themselves software engineers, but anyone that the CCPE and its affiliated provincial and territorial regulatory bodies deem to be practicing software engineering, which is to say almost all practitioners in the software field today. Clearly, the consequences of strictly enforcing such a limitation would be disastrous. On this issue CIPS strongly opposes the CCPE, and asks members to use their resources and contacts to resist CCPE's well financed intent.

Should SE practice ever be restricted? Software engineering still has no commonly accepted body of knowledge. There is an ongoing attempt to define one, known as the Guide to the SWEBOK, initially co-sponsored by the IEEE Computer Society and the Association of Computing Machinery (ACM). But the ACM later withdrew its support because the IEEE intended the SWEBOK to be required for licensing: —the ACM Council decided in May 1999 that it could not support licensing of software engineers. ACM's position is that our state of knowledge and practice in software engineering is too immature to warrant licensing. Moreover, Council felt licensing would be ineffective in providing assurances about software quality and reliability.“ Given a SWEBOK eventually evolves, it will need to stand the test of time - becoming generally accepted through measured efficacy - "faster, cheaper, better" systems. And conversely, experience should document how ignoring the SWEBOK tends to cause harm. Until such a body of knowledge develops and is shown to be essential for competent practice in the field, restricting theory and practice through legally enforced measures is inappropriate and counterproductive, and ironically could harm the public (e.g., by excluding vital expertise).

Where does SE belong? SE's home - computer science or engineering - should depend only on each university's preference. CIPS encourages sharing SE programs wherever possible as the combination of the two disciplines could enhance the educational experience for students and ultimately provide a more qualified practitioner.

How does Canada compare to other jurisdictions? Canada's ongoing disharmony is already causing opportunities for strong SE programs to be lost, human and economic resources in litigation to be dissipated, employers to be confused, the public to be at risk, and our competitive position vis-a-vis global markets to be eroded. Indeed, developments outside Canada have produced the kind of harmonization that continues to elude us. To wit:

- In the U.S.A., the Computer Science Accreditation Board (CSAB) is now a member of the Accreditation Board for Engineering and Technology (ABET) and is the "lead society" for SE programs (as well as Computer Science and Information Systems programs), and is a "cooperating society" for Computer Engineering programs. Complimentarily, the Institute of Electrical and Electronics Engineers (IEEE) is a "cooperating society" for SE programs and is the lead for Computer Engineering. As the lead, the CSAB identifies and trains SE-program evaluators, and sets the SE program-specific criteria, which are managed and administered by the Engineering Accreditation Commission (EAC), and must remain consistent with the EAC's general engineering criteria.
- In the UK, the British Computer Society (BCS) is the only chartered professional institution for the field of information systems engineering. The BCS exists to provide service and support to the IS community, including individual practitioners, employers of IS staff and the general public. The BCS is also an Engineering Institution, fully licensed by the Engineering Council to nominate Chartered and Incorporated Engineers, and to accredit university courses and training schemes.
- In Australia, the Australian Computer Society and the Institution of Engineers, Australia, have agreed to set up a joint Software Engineering Board and this has been implemented. This Board is owned by both societies. In Australia most Software Engineering programs are located in IT faculties with the Dean usually being a computer scientist. There are no such ownership claims over the terms "software engineering" by the IT or engineering people, and the term is used loosely by some IT practitioners who have never been involved in large scale and complex software systems.

In summary, we encourage computer scientists and engineers to work together wherever feasible, to teach and do research in this very important emerging field. More specifically, we encourage current efforts to find a way to jointly accredit SE

courses and programs, and to permit academic freedom in the choice of venues for SE education. Last but not least, while we strongly encourage professionalism through accreditation and certification, employers should be able to recruit and employ whomever they deem best for their needs, to decide for themselves the qualifications necessary in a software engineer without outside interference. Thus, computer scientists as well as engineers should retain both the right to title and the right to practice.