Subject: ACM Endorsements and Concerns

Date: 10/28/98 9:40 AM

From: "Dennis J. Frailey" <d-frailey@ti.com>

To: "Donald Bagert" <bagert@ttacs.ttu.edu>,
"Gerald Burnham, P.E." <burnham@utdallas.edu>,
"William Marcy, P.E." <cswmm@coei.coe.ttu.edu>,
"edd" <dandm@texasonline.net>,
"Otto M. Friedrich, Jr." <Otto_Friedrich@mail.utexas.edu>,
"David Rentschler,PE" <rentsch@austx.tandem.com>
CC: Dennis J Frailey <d-frailey@ti.com>, john.speed@mail.capnet.state.tx.us,
Leonard.Tripp@PSS.Boeing.com

This is a report to the software engineering advisory committee about some recent events of importance to our activity.

Last week the ACM Council considered several proposals related to software engineering. The IEEE Computer Society will consider the same proposals in early November. The gist of these proposals is to formally establish the Software Engineering Coordinating Committee (SWECC), which will act as the representative of both societies in software engineering matters, such as matters relating to licensing examinations. Between them, these two societies represent the vast majority of practicing software engineers and the Texas Licensing Committee has already requested and received support from them in connection with developing the body of knowledge and licensing examinations for software engineering.

This is a summary report of what happened and some of the issues raised. I think they indicate the deep concern among the ACM Council that software engineering be established in a responsible and professional manner.

The ACM Council endorsed the Code of Ethics for
Software Engineers

The ACM Council Approved formation of the joint Software Engineering Coordinating Committee (SWECC), which will represent ACM and the IEEE Computer Society in dealing with software engineering matters.

Issues and concerns were raised in many areas, and I call your attention to these because I believe the Texas Board shares most if not all of these. Note: these are my paraphrasings of what was discussed by the ACM Council. These are not the official positions or statements of the Council, since most of these were not voted on.

Rather, they were communicated to me with the intent that I communicate them further.

Concerns and issues:

- software engineering must be taken very seriously as its own unique discipline -care must be taken to see that individuals who would claim to be software engineers, especially licensed software engineers, have a very solid grounding in software engineering fundamentals. It is not sufficient, in ACM Council's view, for a PE from another field to have been developing software for a number of years. There must be assurance that they are CURRENT in the software engineering discipline and that they be FLUENT in the fundamentals of software engineering.
- In particular, the Council urges licensing bodies to exercise great care not to "grandfather" existing PEs whose backgrounds are in other fields without making sure they are well grounded and current in software engineering as a discipline. (It was noted that some of the greatest software engineering disasters on record were developed by licensed PEs (from other disciplines). This helps make the point that traditional engineering methods, alone, are not sufficient for proper development of software).
- software engineering is a very rapidly evolving discipline and thus it is essential that the body of knowledge and examinations be regularly updated AND that practicing professionals be expected to participate in a serious and thoughtfully defined program of continuing education in order to maintain their credentials
- the distinction between computer science and software engineering must be clearly delineated, although there is a high degree of overlap between these two disciplines
- at the same time, it is essential that software engineering's roots in mathematics be recognized. In some important ways software engineering is distinct from other traditional engineering fields, and this distinction goes to the heart of why it is an important discipline on its own. This distinction must not be lost in the process of establishing software engineering as a professional discipline.
- the fundamentals of engineering examination should ultimately be revised to reflect the fact that software engineering and certain other

engineering disciplines are founded more on mathematics than science and therefore should (at least for any option taken by candidates for software engineering licenses) include stronger emphasis on subjects such as discrete mathematics and statistics and should not depend on topics such as thermodynamics and strength of materials, which are

irrelevant to most software engineering applications. - undergraduate curricula in software engineering might become distorted if students are forced to take courses in topics like thermodynamics and strength of materials in order to qualify for the FoE examination. Since many states are now severely limiting the number of credit hours that a

student may take to get a bachelor's degree, this is a particularly acute concern at this time - a great many computer science programs are not housed in colleges of engineering, and in many colleges these are the logical places to incorporate software engineering programs. Care must be taken that the procedures established for software engineering do not inhibit the establishment of accreditable software engineering programs outside colleges of engineering. (The CSAB model which currently works for computer science, would seem to be a logical starting point for software engineering, and the pending integration of CSAB with ABET should further facilitate this) - Licensing of the individuals who make a product does not, by itself, assure that the product is built correctly. This is especially important for software because it is intangible and its flaws are often very subtle and hard to detect. Thus consideration should be given to product certification in addition to licensing, in order to achieve the

objectives of assuring public health, safety and welfare. - there are many vehicles in place already that might be used as a basis for such certification - for example various international standards for software development and the SEI's Capability Maturity Model for software management and processes.

ACM Council also requests that there be a deliberate effort to guage the support for licensing among industry and other sectors. I have indicated that one such guage is the extent to which industry is willing to fund the development of the body of knowledge and the licensing examinations.

There is also concern about the state laws requiring that faculty who teach engineering be licensed. There are several aspects of this concern: - if they teach software engineering, they need to be competent in software engineering. There is fear that a faculty member in, say, civil engineering might be approved to teach software engineering because he is a licensed PE, whereas a faculty member in computer science who has extensive software engineering background but lacks an engineering degree might not be permitted to teach it

- software engineering should be teachable outside of engineering schools - access to accreditation should be available regardless of what college of the university teaches software engineering

I think this is a fair summary of the ACM Council's concerns. The ACM Council includes some of the most highly respected computer and software specialists in the world. I am pleased to note that their concerns mirror many of those we have discussed in our deliberations with the licensing committee.

Regards, DJF

Dennis J. Frailey d-frailey@ti.com
Senior SW Technologist 972-344-5607 (voice)
Raytheon Systems Company 972-344-5602 (fax)
POB 655012, MS 49
Dallas, TX 75265