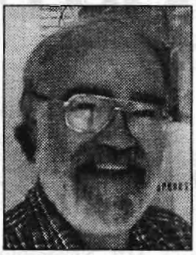
The ethical software engineer

BY DON GOTTERBARN
ACM Ethics Committee Chairman

The Association for Computing Machinery (ACM) and the IEEE-Computer Society have adopted the Software Engineering Code of Ethics and Professional Practice (5.2). This is significant, given that the state of Texas has already licensed software engineers without defining any standard of ethical practice. This oversight is dangerous for two reasons. First, licensed software engineers will obey the law, but laws provide inadequate guidance in many critical situations. And second, the failure to connect specific ethical standards to licensing encourages the mistaken view that there is little agreement among software engineers about their professional and moral obligations. The code's development, however, indicates a significant agreement among software engineers about the way they ought to behave.



Don Gotterbarn

The code also provides mechanisms to help practitioners make ethical judgments in those situations where the law is silent.

The code was developed by the joint IEEE-CS/ACM task force on Software Engineering Ethics and Professional Practice (SEEPP). The SEEPP task force is multinational in citizenship and in membership in professional computing organizations. After extensive study of several codes of ethics of computing societies, engineering societies and other professions, SEEPP selected imperatives for the draft code. SEEPP also contributed new imperatives related to its knowledge of software engineering and based on external reviewers' suggestions.

The draft code was reviewed by members of several professional computing societies and went through several revisions. Version 3 appeared with a turnaround ballot in the IEEE-CS's and the ACM's flagship magazines. Most clauses received better than a 90 percent approval rating. Contributed comments led to the development of Version 4 which SEEPP submitted for peer review using the IEEE's formal technical standard review process. Again, the code easily passed this process. Comments were used to develop the final version of the code (www-cs.etsu.edu/seeri/secode.htm) which was approved by the ACM in November and the IEEE-CS in December.

I found the consistently high level of agreement about the behavior expected of a professional software engineer very significant. There is general agreement about our obligations as software engineers, even if some software engineers give in to external pressures not to follow these obligations.

The code aids decision making by overcoming two difficulties with other codes. First, most codes of ethics provide a finite list of principles which are often presented as a complete list and the reader presumes that only things on the list should be of ethical concern for the professional. Second, many codes provide no guidance for situations where rules, having equal priority, appear to conflict. This equal priority leaves the ethical decision maker confused. The software engineering code addresses both of these limitations.

The code explicitly rejects the concept of completeness.

"It is not intended that the individual parts of the Code be used in isolation to justify errors of omission or commission. The list of Principles and Clauses is not exhaustive. The Clauses should not be read as separating the acceptable from the unacceptable in professional conduct in all practical situations. The Code is not a simple ethical algorithm which generates ethical decisions."

The code addresses completeness by providing general guidance for ethical decision making, especially in those areas not explicitly mentioned in the code.

"Ethical tensions can best be addressed by thoughtful consideration of fundamental principles, rather than blind reliance on detailed regulations. These Principles should influence software engineers to consider broadly who is affected by their work; to examine if they and their colleagues are treating other human beings with due respect; to consider how the public, if reasonably well informed, would view their decisions; to analyze how the least empowered will be affected by their decisions; and to consider whether their acts would be judged worthy of the ideal professional working as a software engineer. In all these judgments, concern for the health, safety and welfare of the public is primary; that is, the "Public Interest" is central to this Code."

The first principle asks the developer to consider all stakeholders, not just the software engineer's employer or client. The second principle—due respect—requires a protection of human values. This section states that in all decisions the public interest is the primary concern.

To reinforce the priority of public well-being, the code asserts the priority of concern for the public over loyalty to the employer or profession. It is a professional's obligation to take positive action to address violations of the code. The code addresses both the responsibilities of the practicing professional and of the profession. Several large software companies have posted the code as an expected standard for their employees. Its adoption by two large computing organizations is a positive step because this code is not designed to be self-serving to the profession. The code requires software engineering professionals to be ethically responsible to all of those who are affected by their products.

The IEEE Ethics Committee maintains a Web site at "www.ieee.org/committee/ethics". The author can be reached via e-mail at "gotterbarn @Access.ETSU.edu".