

Interviewee: Steve Barber (SB)
Interviewer: Michael Davis (MD)
Location: New York City
Date: 11/14/2002 *approved*)

QUESTIONNAIRE

SOFTWARE DEVELOPERS WRITE A CODE OF ETHICS

1. What is your educational background?

S.B., MIT 1987. Computer Science and Engineering. Humanities concentration in Science, Technology, and Society

J.D., Benjamin N. Cardozo School of Law, Yeshiva University 1993.

2. What sort of organization do you work for? What do you do there?

My employer is a software consultancy whose clientele consists entirely of financial institutions and firms, primarily in the capital markets. Our projects almost always use Java as the implementation technology; this has historically been our niche.

My current assignment is with a securities clearing corporation. I began this assignment in February 2002. I am there to provide mentoring in Java and related technologies, to review and improve the Java-based systems architecture, and to design and implement some components of that architecture. The system itself is a Web-based front end for trade entry and inquiry as well as supporting functions.

My prior assignment was as the technical project manager for a trader workstation and trading exchange for complex commodity derivatives. My role was to meet with clients, gather and analyze requirements, specify the system architecture and detailed technical design, prepare a project plan with resource and cost estimates, lead an eight person development team, and do small parts of the implementation. This assignment lasted two years.

It is worth noting that during the time of my involvement with the SEEPP, I was a practicing attorney, focused on software and online intellectual

property issues.

3. What experience, if any, have you had in software development?

I have been developing software professionally for thirteen years, from 1984-1990 and 1995-present. The first six years I worked as an in-house developer for a vendor of packaged commercial software. We made IBM protocol emulation packages for PCs and Unix-based systems. I was exposed to and performed tasks related to almost all phases of the software development lifecycle — module design, implementation, testing, in-house and field maintenance, and project management. I also participated in pre- and post-sales customer support. During the last seven years, I have been a software consultant to Wall Street firms. Much of this phase of my career has been in project and practice management, with a liberal amount of architecture and design, only more recently spending significant amounts of time doing hands-on development. In the late '90s, I did a lot of pre-sales work, presentations, and co-wrote and edited a technical book (Lewis, Barber, and Siegel, *Programming in Java IDL*, Wiley 1987).

If you are a software developer, what led you into that field?

As a sequence of events, my father first introduced me to programming while I was in high school. He taught programming himself, and brought home one of the school's Radio Shack TRS-80 Model I computers for me (and him) to work with on weekends. Soon I was writing games and other programs, and even had a part-time job programming while other kids were flipping burgers. I knew that I wanted to pursue software in college and as a career direction from high school, and never seriously questioned that path.

As a matter of what aspects of software development I found attractive, it was probably a matter of the logical foundations and sheer determinism of programming. I could use what I knew and was good at to get things done, with fairly immediate feedback. There wasn't much else like that for me. And I have always enjoyed the process of programming as much if not more than the end result.

Since I have a bit of an unusual career path, having started in software, leaving it for law, and then coming back, it might be worth a couple sentences to discuss what led me back. While I enjoyed law school and

studying the law, I found that law practice was mostly tedious and difficult for me. Worse, the clients wanted me to be aggressive and mean for them. Worse still, many clients complained bitterly when we charged them money to fix nasty situations that they, with just a little foresight, easily could have avoided. Worse, these situations frequently arose after they ignored our previous advice to pay us (much less) to do preventative or strategic legal work. Software development has few of these distasteful aspects, or at least they occur with far less frequency. Even in consulting, where clients can be demanding and unrealistic and frequently change their minds, working in the software field is much more satisfying. In short, I found that I much preferred building stuff to fighting about stuff.

4. Are you an engineer?

I don't think so, though this is a very interesting question, and the answer is not at all clear cut.

5. How did you hear about the IEEE/ACM Joint Task force on Software Engineering and Professional Practice (SEEPP)?

In 1994, I was working as an associate to an attorney named Lance Rose. Lance was involved in the Computers, Freedom, and Privacy conference. Someone on the program committee asked Lance if he would speak on a CFP panel as someone opposed to the notion of licensing software professionals. Lance had previously spoken on this topic to a software consultants' group in New Jersey. Lance was doing other things at the conference and didn't really feel like he had much more to say on the topic, so he suggested I do the licensing panel instead, especially since I actually had a software background and he did not.

That panel's organizer was a fellow named Don Gotterbarn, who needs no introduction here. After I spoke for my ten minutes on all the evils (Lance and) I could think of regarding software professional licensing, Don invited me out to dinner, laid out his thoughts on the subject, and invited me to join the SEEPP and work on the software code of ethics.

I have often wondered why Don would have invited someone who had just "gone public" against the professionalization of software engineering to work, essentially, for such professionalization. I have several theories. One, having actually taken the time to research the topic and speak about it, I probably was as qualified as any software practitioner on the topic.

Two, as a professional software developer who was also a practicing attorney, I had the fairly rare experience among software people as actually having to live under the attorneys' rules of ethics, for better or worse. Three, SEEPP was fairly heavy with academic ethicists, but light on actual programmers. Four, it often pays to keep ones friends close, and one's enemies closer.

Not that I have ever truly been an enemy of ethics codes — at worst I merely have commented upon them cynically. My dissatisfaction with the legal ethics rules stems from my belief that in practice they 1) do little more in practice than define the ethical envelope against which many attorneys do their best to push, and 2) provide a locus for statutes or regulations that basically codify professional protectionism against accountants and paralegals, often artificially inflating the cost of legal services, depriving those of modest means from being able to afford them.

6. What lead you to participate in SEEPP's work?

Besides Don's kind invitation, I looked forward to a chance to participate in something that could shape, hopefully for the better, the software field for many years to come, and under which I might someday have to live and practice. Perhaps I would have a chance to eliminate or temper unrealistic or onerous provisions. Perhaps I might help software engineering avoid some of the nastier so-called ethical burdens under which the legal profession has burdened itself. Working on SEEPP was also a professional networking opportunity. I also thought that a lawyer might have something to add by applying legal drafting skills to the whole resulting documents and trying to make them more self-consistent.

Further, I thought that a lawyer's perspective was useful to the SEEPP drafting process in another way. Many of the committee members were working from an ethicist's standpoint, trying to construct a document that worked logically and coherently as a matter of ethics. They were trying to create a work that sought to capture and transmit values. The legal viewpoint might help make that lofty goal a bit more practical: helping to create a document that would not only express values, but that could also be interpreted for and applied to concrete situations. I hoped to apply the lawyer's eye to help reduce unintended ambiguity, and to use language that was consistent with some of the legal language relevant to areas of law that might come in contact with the code (for example, malpractice

liability and standards of care.

7. Were you familiar with codes of ethics before you became involved in SEEPP? Explain.

I was familiar with the American Bar Association Model Code of Professional Responsibility and the ABA Model Rules of Professional Conduct, as well as their enactments (with variations and frequent modifications) in New York and New Jersey. My familiarity with them was that I completed the required legal ethics course that covered these codes while in law school, studied them to prepare for and pass the Multistate Professional Responsibility Exam, and practiced law for two years having to consider them from time to time to govern my actions as an attorney.

8. In what ways did you participate in SEEPP's work, especially in the process of preparing the code? (The more details, the better.)

As it turned out, I didn't participate actively for very long. I recall attending two SEEPP meetings in Washington, and perhaps one in Philadelphia. This was fairly early in SEEPP's work, just after being organized. I contributed most actively not to the code itself, but to the SEEPP Task Force Guide to Operations, a set of procedural rules governing the operation of SEEPP itself. Such a document was required by IEEE process and was derived from a model Guide. The SEEPP Task Force met and revised and modified the model Guide to be appropriate to the concerns and structure of SEEPP. I cast a lawyerly eye on the procedures in the guide, and I think helped make them less ambiguous and a bit more practical. The task force member who acted as secretary at one of the meetings lost his notes (computer failure), and I reconstructed the Guide changes from the notes I had taken at the meeting and forwarded them to Don. Also during these meetings, the task force sketched out what became the high-level principles of the code. I was assigned to a working group for one of the principles, but I don't believe that the working group ever met by the time I told Don that I wouldn't be actively participating in SEEPP any more.

I stopped paying much attention to SEEP sometime in 1996. I switched back to software from law in September 1995. The career change left me with less time for SEEPP and other "extra-curricular" activities — I had to focus on learning a lot of software technology that I missed during my five-year excursion into the law.

There were some other things that added to my willingness to de-prioritize my SEEPP work. The SEEPP task force was dominated by academic ethicists -- many among these were people who basically taught computer ethics courses in colleges for a living. First of all, among these people I was basically out of my league. They had spent years reading and developing the computer ethics literature; I could barely articulate what I thought the word ethics meant. Second, I was somewhat concerned about the relative lack of practicing computer scientists and software engineers on the task -- it seemed to me that while ethicists could perhaps catalyze and facilitate discussions about software engineering ethics, there needed to be a stronger component coming from actual engineers who faced and ultimately had to deal with the ethical issues that arose in practice.

Third, no matter how well meaning and competent the task force members appeared to me, I couldn't help wondering about an appearance of a conflict of interest: if the code were adopted and then ultimately incorporated into state-mandated licensing programs, didn't the task force members stand to benefit directly by the increased demand (or at least increased job security) for computer ethics professors? Let me stress that in raising this issue, I in no way wish to even imply that anyone on the task force actually acted with such a motive in mind; I like and even admire the task force members I worked with and believe that they sincerely wanted to better the world by helping to draft the code. This was, however, an issue in my mind at the time and I still see it as a structural weakness in the process of drafting the code.

In the face of decreasing available time, I found all of these factors together led to me dropping SEEPP from my "active projects" list.

- 9. By what means did you participate? For example, did you participate by email, or by phone, or through face-to-face meetings, or by letter, or by informal conversation, or the like?**

I monitored the se-ethics e-mail list, but rarely if ever posted to it. I had several phone conversations with Don Gotterbarn, and attended I think three face-to-face meetings.

- 10. Did any of these means of participation seem to work better than the others? Any seem to work worse? Which would you recommend as best? Why?**

The face-to-face meetings were the best for me. They focused my attention on SEEPP for the time I was there, for one thing. I enjoyed the direct interaction with my colleagues on the task force. Face-to-face is also better for coming to consensus or compromise positions than e-mail; people are typically a little less aggressive in person than when typing away by themselves at a computer.

E-mail wasn't effective for me; they either didn't happen for months at a time or else there were many long ones seemingly all at once that I never got around to reading. This is odd since I am often an avid reader and writer of e-mail.

11. Any events that particularly stick in your mind relevant to the process? (The more details, the better.)

I remember a telephone conference with an IEEE process guru that seemed to go on forever with no real benefit. SEEPP members familiar with the process guru engaged in quite a bit of eye-rolling.

I remember it was quite a long time from when Don invited me to participate until anything much seemed to happen. I remember long stretches between meetings where no one seemed to do much. But eventually, the group gelled and started actively drafting.

12. Do you have any documents, paper or electronic, relevant to your participation in the process? May we have a copy?

Yes, and yes, if I ever find them.

13. Has your thinking about codes of ethics changed as a result of your participation in SEEPP's work? How?

No.

14. What, in your opinion, is important about having a code of ethics?

A code of ethics can provide knowledge and guidance to new and experienced practitioners. It can provide a "safe harbor" for practitioners beyond which no one need tread. It can act as a reminder of higher

aspirations.

15. Is there anything about your *participation* that you are especially pleased with or unhappy about?

I'm pleased I was able to improve the Guide to Operations. I have some regret that I didn't stick with the Task Force through the issuance of the first version of the code. While I have some reservations about a few of the clauses, all in all it is an admirable document.

16. Is there anything about the *final code* that you are especially pleased with or unhappy about?

I'm pleased that the code was actually completed and then adopted by both the ACM and the IEEE. There were times when it appeared that not much was going to happen.

I'm pleased that for the most part the code captures many values to which software engineers would naturally aspire.

I'm bothered that much of what has been written about the code claims that the code puts the welfare of the public as its first priority but that the code does not actually state this except in the preamble. Preambles are usually not enforceable.

I'm bothered that under the code a software engineer is required to report to his employer about intellectual property violations. I have never worked for an employer or client that actually had all their software licensing in order, either intentionally or unintentionally. Including such an unenforceable provision that automatically makes many practitioners unethical seems to me to undermine the authority and reasonableness of the code more than it would admonish or instruct the practitioners (who would for the most part be unemployable should they actually conform to the code in this matter).

17. Is there anyone whose participation in the process seems to you especially important? Explain.

I think Don Gotterbarn was really the driving force behind the code effort, and ultimately kept it moving when the task force's momentum appeared to be slacking off.

18. Anyone who you think we should be sure to talk to? Explain.

No.

19. If you had been in charge of the process, what, if anything, would you have done differently?

No.

20. Is there anything we should have asked but didn't? Anything you want to add to what you have already said?

No.

Center for the Study of Ethics in the Professions
Illinois Institute of Technology
Chicago, IL 60616