from The Rational Edge: The ethical issues involved in managing and developing information technology are many, and they are increasingly complicated by the power of individuals and infrastructures. Read how a college professor discusses ethics with his students as they prepare for careers in IT.

This is a time of year when thousands of students are graduating from colleges around the country and hoping to land the job of their dreams. Computer science graduates interview in many types of companies for jobs in network administration, quality engineering, programming positions, field engineering, and other critical positions. Many enter the job market looking for their first real job in their chosen field, while others have been working as summer interns for several years and already gained practical experience.

As a college professor, I can say with confidence that we prepare the students for the technical challenges of the many jobs in IT. But I wonder sometimes whether we do as good a job preparing them for the rest of what they'll face.

How many of you have had any training in ethics, specifically the ethics associated with technology and its applications? I know I never did. At Worcester Polytechnic Institute we have a course called Social Implications of Information Processing. Most of our undergraduate computer science students take the course. However, very few graduate students take the more advanced version of the course. Undergraduates must take the course, while graduates can elect to take it. I did a quick search for similar courses at other schools and was pleasantly surprised to find several that offer one.

After preparing to teach the course to graduate students, I'm convinced that some training in ethics can benefit everyone involved in software development. I'd like to look at some of the topics that
are covered in the social implications courses. I'm going to use the framework from a wonderful book, *A Gift of Fire*, 2ed. by Sara Basse.  

The ethics landscape

First, let's define what we mean by ethics. The primary definition in the Merriam-Webster online dictionary says, "The discipline dealing with what is good and bad and with moral duty and obligation." One of the secondary definitions says, "The principles of conduct governing an individual or a group." I realize that there are different views of morality and we could go down a very slippery slope discussing them. So, decide upon what you consider "moral duty and obligation" before reading further. I suspect most of us will have similar views.

In practical terms, those who work in IT will likely be responsible for implementing or planning technical components that partly shape their organization's ethical dimensions. I'd like to consider a few of these areas, especially ones that have been in the news for the past five years.

Privacy

Three issues dominate a discussion of privacy:

1. Freedom from intrusion
2. Control of personal information
3. Freedom from surveillance

I think of Orwell's classic, *1984*, when I ponder these issues. Look around and you can see daily examples of unethical conduct with respect to each of the above issues.

Freedom from intrusion means that we have the right to be left alone. U.S. citizens expect this right as a part of their citizenship. Yet, we have to regularly deal with telemarketers, spam, and many other forms of intrusion into our lives. Every time we think we've eliminated an intrusive practice another emerges. Do you get as annoyed as I do when I pay to see a movie and have to sit through ten minutes of advertisements for businesses and products that have nothing to do with the motion picture industry before the movie begins? This is a form of intrusion, and I pay for the experience. I wonder if there will be an enterprising theater that will charge a little more for "advertisement free" theaters.

Do you control your personal information? If you do, you are one of the rare people who does not own a computer, has no credit cards, never pays for anything in installments, doesn't use a cell phone ... It's impossible today to know where all of your "personal" information resides, much less control it. The amount of information about an individual is a phenomenon that has exploded in the last fifty years of our information technology revolution.

Freedom from surveillance is the final privacy issue. How often do you remove spyware from your computer? Enough said -- well, almost enough. There are so many types of surveillance today that it is all but impossible to avoid them all. Of course, one might ask whether the satellite images that show my house are an invasion of my privacy.
When we discuss the ethical issues of privacy in the information age, we have to ask: Is privacy possible? If it is, then how do my actions either protect or remove the possibility of privacy?

Encryption

Encrypting information is an ancient practice. We regularly encrypt electronic communication today because of the proliferation of identity theft and other information-based crimes. Encryption is one way to guard privacy. If I control my personal information, I want to ensure that I am the only one who can access that information. If someone maliciously accesses it, I want to prevent them from reading it.

This seems like a simple, reasonable goal. Other issues, however, come into the mix. In 1991, a bill was drafted in the US congress that would require telecommunications providers to ensure that the government could obtain unencrypted forms of information that are transmitted over their channels. That bill did not pass, but there have been others that did pass and today, the government has the ability to intercept, access, and read or hear the content of most communication. If the US government can do this, can other presumably less scrupulous parties do the same thing? I believe we would be naive to think they cannot. This is the crux of the issue with respect to ethical behavior for industrial software developers. What is our responsibility for ensuring the integrity of the information in our systems?

Unfortunately, I cannot guarantee the "safety" of my clients' information and communications that pass through my systems. The only thing I can provide is due diligence toward protecting their information to the best of my ability. Time and again we have seen security methods thwarted and broken. If there were a perfect encryption technique, I'm sure there would be many legal barriers to its use. Encryption issues take us more into the political and social arenas. Humans have devised means to transmit secret messages for millennia. Governments -- especially their military branches -- rely on secure information transmission in order to achieve their objectives and preserve and protect their constituents.

If we accept that governments and others have legitimate reasons for secure transmissions, then we must accept that codes and other encryption techniques are legitimate and ethical. We must then consider whether developing systems to break encryption schemes is ethical or not.

Trust

When my doctor tells me that my cholesterol is too high and prescribes either a change in diet or medication, I pay attention to him. Why? Because I trust him. I believe he is an ethical person who looks out for my best interests. When the computer tells me that I need to install a new or updated program, or accept a cookie through my Web browser, I may stop and think about it, but usually will accept the advice without possibly understanding the full implications. If I accept a malicious piece of code that wreaks havoc on my computer, whose fault is it? I am certainly guilty of naiveté the creator of the malware (another term we've come to know all too well) is guilty of unethical behavior.

Sadly, even when we are sure that the software was not created with malicious intent, we are susceptible to trust problems. The doctor I trust generates his bills through a computer system. I
trust the system to bill me fairly. However, the bill may be wrong. If the bill is a little bit wrong, I may not question it. If it is off by a lot, I will certainly question it. Who has the responsibility for ensuring that the bill is correct? And is this an ethical issue?

The question has to do with harm. Most codes of ethics for any profession address the need to avoid harming others. When software fails to behave in a trustworthy manner, harm often results. In the case of the billing error, the harm is financial, and usually minor. But, what about software that controls medical devices or aircraft? The harm that can result from malfunctions might be the loss of life.

Harm can be inflicted intentionally or unintentionally. In most cases, intentionally inflicting harm is considered wrong. Civilized countries reject the practices of torture because they consider it immoral. But most of us would agree that a doctor who amputates a limb in order to save a patient's life acts ethically. A doctor who performs unnecessary surgery on a patient in order to increase payment from the insurance provider acts unethically.

Unintentionally inflicting harm is where the waters become murky. Doctors are human and can make mistakes in diagnoses or treatments that result in harm to their patients. Software developers release software containing defects that cause harm. We have come to accept that most of these defects are unintentional and often take precautions to limit the harm, such as daily backups of our data.

The ethical approach to the potential for harm, in my opinion, involves taking responsibility for your work and acting appropriately. One of the reasons that heart monitoring software costs so much more than the software that schedules your child's soccer games is that the negative effects of a system failure are so much greater in the first instance. A responsible, ethical software developer will spend much more time ensuring the quality of the heart monitor software -- and rightly so. If we are going to trust software, we must trust the people who build it.

**Freedom of speech**

Regardless of the country you live in and the degree to which you enjoy free speech as a protected right, when you're on the Internet, you probably feel a little more freedom of expression than in the work place or in public transit. But that perception can be dangerous. We see employees dismissed because of something they've written in their blogs. In worse cases, we see citizens jailed or deported for criticizing their governments in cyberspace. And as an unusual byproduct of the general freedom many writers feel free to "say anything" on the Web. Some casual researchers will cite Websites as authorities without checking the credentials of the people writing for those sites.

Many societies throughout the world value freedom of speech. On the surface, it seems that the only ethical issue might regard the suppression of this freedom. But consider spam: Do spammers have the right to subject others to their views? Every day we are bombarded by new forms of spam in email and on Websites. Some spam is commercial, some political, some just annoying. The one thing all forms of spam have in common is that they are unwanted. Is it ethical to send spam? Is it ethical to prohibit the delivery of spam?
These questions ask us to consider freedom of speech. If you are interested in seeing how one case played out in court, I recommend you look at the Intel vs. Hamidi case. In this case, a former Intel employee, Ken Hamidi, sent a mass email to over 30,000 Intel employees. Hamidi was dismissed from Intel in 1995 and, in 1998, sent the mass email that discussed the company's labor practices. Intel wanted to block Hamidi from this practice. They argued that Hamidi did not have the right to intrude on Intel's property by using its equipment to deliver his messages. Intel likened the actions to trespassing. They said Hamidi trespassed on Intel property when he sent his email to the company's employees. Hamidi, in turn, said he had the right to send the email from his own computer and that the issue was one of free speech, and not trespassing.

The court ruled for Intel. The judge said that the lawsuit was a case of trespass and that the service provider, Intel, has the right to get a permanent injunction to block e-mail sent by Ken Hamidi. He offered no further explanation of his ruling. The ruling was subsequently overturned by the California Supreme Court, which said that Intel could not prove there was any damage to their computers as a result of the email; therefore, there was no trespass. The court stated: "any more than the personal distress caused by reading an unpleasant letter would be an injury to the recipient's mailbox, or the loss of privacy caused by an intrusive telephone call would be an injury to the recipient's telephone equipment." Where do you stand on spam issues like the one above? What about spam creating programs? Is there a difference? Consider a spammer who writes a program to send unwanted email messages to recipients on a regular basis. Is this any different from an email marketing company that regularly sends email to addressees from a list they purchased? Your ethical and moral values will affect the way you answer the questions. This is a key point. If ethics and morals go together, you have to hone your skills at reasoning critically about ethical and moral issues. Some religious traditions call this having a well-formed conscience. Whatever you call it, you need to discuss and debate the relevant issues and come to well-formed conclusions. You probably already realize you need to do this as an individual, and as an IT worker, you need to do this as part of your role in an organization. Corporate citizenship should be based on a sound ethical as well as business foundation. Too often, our decisions are based on emotional responses to an issue. In fact many marketing, political, and media organizations count on just this response in order to sell us something or convince us to vote their way.

**Intellectual property**

Intellectual property represents one of the most important ethical issues today: that of ownership, and the value and respect which that ownership entails. Most commercial enterprises in the technology sector probably spend more time on it than all of the other ethical issues combined. The issue of intellectual property ownership sharply divides the software development community because software is, in fact, intellectual property.

Software is intangible. It is the embodiment of an idea. We can think of software like we do writing, or music, or paintings. Each of these media have legal protection under copyright law. Should software be equally protected? What exactly does this protection afford?
Copyright law, like most law, is complex. One of the complexities is the concept of *fair use*. Fair use means that one can use copyrighted material when it contributes to the creations of new work and does not deprive the authors or publishers of income for their work. An example is quoting part of protected material in a review.

The intellectual property and copyright issue is central to the free software movement. Richard Stallman, the most vocal and visible advocate of free software says that free software is really about several freedoms:

- The freedom to run the software
- The freedom to study the software by viewing the source code
- The freedom to redistribute copies of the software
- The freedom to change the program and release your changes for free.

If you have ever heard Richard Stallman speak about free software and his views about software, you were probably struck by his passion. You may not agree with his arguments -- I don't agree with all of them -- but you will have an understanding of what he considers moral and just. He uses words like "moral" and "evil" when he presents his views on software ownership. If you read the "Philosophy of the GNU Project," you will understand the fundamental principles behind the free software movement. The paragraph that stands out to me is: "Free software is a matter of freedom: people should be free to use software in all the ways that are socially useful. Software differs from material objects -- such as chairs, sandwiches, and gasoline -- in that it can be copied and changed much more easily. These possibilities make software as useful as it is; we believe software users should be able to make use of them."

Software is, in fact, socially useful. Further, it is economically useful, just as chairs, sandwiches, and gasoline are. You have to decide where you stand on whether or not software should be owned. My personal belief says that one should be able to decide if they want to own their work -- whether it is physical work, like wood carvings, or intellectual work -- or not. Those who decide to maintain ownership are ultimately going to let the commercial marketplace decide their degree of compensation. Those who offer their work for free have decided that it is more important to share their ideas and perhaps reap monetary rewards after the fact from customization or collaboration. I respect both approaches to ownership.

One problem that I see quite frequently, especially with students, is that they become caught up in the idea of free intellectual property and they fail to see a problem when they copy something that is not free. Recently a student and I were talking about one of our favorite television programs. I mentioned that I had missed the latest installment. He said to me, "I can send it to you." When I asked whether the copy was legal he said that a friend of his had recorded it, digitized it, and sent it to him.

I pointed out that distributing the copy was not ethical. His counter-argument was that if he were home he would have recorded it himself, so there was no difference. In his mind, the end justified the means. I refused his offer and we spent a little more time discussing the issue. He admitted that he did see my point, but I'm sure he didn't rush home and destroy his copy. I had the feeling that his view was, "no harm, no foul." No one was being harmed by his actions.
Such attitudes were involved in the famous Napster case, where Napster users were accused of sharing music files illegally. The case examined Napster, the users, fair use, and responsibilities. The courts ruled that Napster encouraged copyright infringement and caused the original Napster Website to shut down.

**Ethics in practice**

Most engineering disciplines address the issue of ethical conduct by practitioners. Many engineers are required to obtain professional certification in order to practice in their field. What about software engineers? There have been several certification attempts for programmers. In the 1970s I became a Certified Data Processor. This certified that I understood the fundamental concepts of the state-of-the-art in programming at the time. I took the certification exam, simply because I enjoy taking tests. The certification neither helped nor hindered my career.

The two major computer societies, the Association for Computing Machinery (ACM) and the Institute for Electrical and Electronics Engineers (IEEE) Computer Society both have codes of ethics by which their members agree to abide. I'm not sure that most of the members know the contents of the codes, and I would not be surprised to find out that most members haven't even read the codes. The IEEE CS also has a Certified Software Development Professional certification (CSDP) program. The CSDP has its own code of ethics and professional practice.

I cannot say how successful the certification programs are in other professional engineering disciplines, but they have had very limited success in software development. For example, there are currently less than 600 people who have received the CSDP status. There are many possible reasons for the lack of success, but let's just assume that certification doesn't work yet in our profession. Let's also assume that there are a significant number of people who belong to ACM and IEEE CS who are not aware of the contents of the code of ethics. Further, the majority of software developers probably do not belong to the ACM nor IEEE CS.

The different codes of conduct have several points in common. The ACM code has 24 imperatives for personal responsibility. The IEEE code has ten points.

The phrase "do no harm" does not appear in the Hippocratic Oath, although most people think it does. While it does not appear in the ethical codes under examination, the essence of the phrase comes through. The ACM code says "avoid harm to others." It elaborates the statement to say that harm "means injury or negative consequences, such as undesirable loss of information, loss of property, property damage, or unwanted environmental impacts." The IEEE code has a broader statement: "to accept responsibility in making decisions consistent with the safety, health and welfare of the public, and to disclose promptly factors that might endanger the public or the environment."

The concept of causing no harm to others is quite a general concept, and one that we find in most professional codes of conduct. The devil, however, is in the details and many grey areas exist. We have already touched on the issue of harm in our discussion of trust. Just as medical professionals have to align their moral compass with the needs of society, software developers must decide...
what is morally right to them. They must decide upon a definition of harm that reconciles with their concept of morality.

There are different types of harm. One is caused by possible negligence of the software developer. Another involves possible negligence, but adds another party -- the malicious hacker.  

Where do software developers hone their ability to reason about ethics. One hopes that it begins with their course work. However, we need to continue to discuss and debate the issues in industry. We need to be clear about what we expect from our colleagues and what we will not tolerate. Industry needs to work with the academic community to ensure that the ethical debate continues inside and outside of the ivy-covered walls.

In closing, I would ask: "What do you expect from your software developers with respect to ethical behavior?" Have you ever thought about it? Does your organization have a code of ethics? If not, why not? When you hire someone, do you ever probe their values for what they consider ethical behavior? Do you assume they share the same values as you do? Perhaps spending a little time, either in a group meeting or some other forum might provide food for thought and help your organization focus on the core values that you want all of your employees to share.

Notes

1 The course description can be found at: http://www.cs.wpi.edu/Ugrad/Courses/cscourses.html.


4 The fourth amendment of the U.S. Constitution protects the citizen's right to privacy.


7 http://www.gnu.org/philosophy/philosophy.html


10 See http://www.computer.org/certification.
11 See http://www.computer.org/portal/site/ieeecs/menutem.c5efb9b8ade9096b8a9ca0108bcd45f3/index.jsp?&pName=ieeecs_level1&path=ieeecs/education/certification&file=ethics.xml&xsl=generic.xsl&.

12 As a side note, when I was looking for this number, I found out that one of my colleagues who sits two doors away from me received his certification in the Fall of 2006. As far as I know, no one in our department is aware of his certification.

13 Some call this person a cracker, in order to differentiate between the good connotation of hacking (used by Stallman and his community) and those who are trying to cause harm.

© Copyright IBM Corporation 2006
Trademarks
(www.ibm.com/developerworks/ibm/trademarks/)