The IEEE/ACM Code of Ethics History

Don Gotterbarn, Software Engineering Ethics Research Institute.
http://seeri.etsu.edu

Ethics

- Also known as moral philosophy, is a branch of philosophy which seeks to address questions about morality; that is, about concepts like good and bad, right and wrong, justice, virtue, etc.

Wikipedia
Content

1. Why should a profession have a Code of Ethics?
2. Chronology
3. Levels of Professional obligation
4. Issues and concerns
5. Professional Tensions during Code Development

IEEE - Institute of Electrical and Electronic Engineers
ACM - Association for Computing Machinery


Why should a Profession have a Code of Ethics?

1. Positive stimulus for ethical conduct,
2. Inspire confidence of customer, user,
3. Educate practitioners, students, managers, rule makers,
4. Educate the public,
5. Provide support to take positive action,
6. Counter pressure against other’s urging to act in ways inconsistent with the Code,
7. Means of deterrence and discipline
   - e.g. revoke membership, suspend/revoke license, fire from a job.
8. Used as a foundation for litigation,
9. Enhance profession’s image,
Chronology of the IEEE/ACM Code

- **Draft 1**: delivered to IEEE-CS/ACM Steering Committee December 1996.
- **Draft 2**: widely circulated for comment January through March 1997. Published in SIGSOFT and SIGCAS bulletins in July 1997.
- **Draft 3**: circulated to industry and other professionals and then published with a turnaround ballot in November 1997 Computer and Communications of the ACM.
- **Draft 4**: revision based on comments and ballots regarding version 3. Submitted to Steering Committee in December 1997.
- **Draft 5**: passed a complete IEEE formal technical review process in September 1998.
- **Draft 5.2**: passed legal review, approved by the ACM November 1998 and the IEEE Computer Society December 1998.
- **Published**: in 1999

Cumulative Levels of Professional Obligation.

Elements unique to particular professional practice e.g. Ensure adequate testing....
Issues and Concerns during the Development of the Code

1. **Why our own Code of Ethics?**
   - Most other professions operate under explicit ethical standards stated in profession-unique codes of ethics.

2. Should the Code be interpreted as a legal document (e.g. to penalize divergent behaviours) or as a document intended to inspire good practice?

3. Can the Code be used to guide professionals in their decision-making during software development?

4. Can the Code be used to alert practitioners to those things for which they are accountable?

Professional Tensions during the Development of the Code

1. **Two approaches to ethics: Virtue Ethics and Rights/Obligations**
   1. **Virtue** ethics holds the optimistic view that if people are simply pointed in the right direction, their moral character will guide them through ethical problems.
      - Code that is inspirational with minimum details, autonomy of judgement
   2. **Code spelling out precisely rights and responsibilities**
      - Rights/Obligations folks used a legalistic model to evaluate each imperative.
      - Include a standard of measurement for each imperative
      - E.g. X tests needed to ensure adequate testing

   • **Solutions**
     1. A Preamble provides some directions for ethical decisions
     2. Code should not be read as complete descriptions or legalistic statements
     3. Principles supported with clauses giving examples
     4. Guidance is provided in selecting between conflicting principles
        - Principles are stated in order of priority
        - Public always comes first before employer or profession
Professional Tensions during Code Development

2. Discomfort with the rules
   1. Some felt powerlessness
      • e.g. ‘Ensure an appropriate methodology is used …’
      • Only management could enforce such a rule
   2. Some wanted specific standards imposed
      • To gain economic advantages, ‘first to the market’, may justify abandoning standards
   3. Some wanted to strengthen the rules
      • e.g. ‘Take responsibility for detecting, correcting and reporting significant errors’
      • Flaw: ‘I found lots of errors but I didn’t think any of them was significant’
      • e.g. disclosing dangers created by software
      • Wording is ‘actual or potential dangers’ to prohibit someone from not reporting a danger because it was not yet real.

Professional Tensions during the Development of the Code

3. Interaction between technical and ethical standards.
   • To include specific standards or best practices into the Code
     • e.g. ‘Path testing be done for cyclomatic complexity greater than 12’
   • Resolution: Rejected
     1. Since as standards improve, the Code could become obsolete, revisions take time (i.e. minimum of a year)
     2. Stating a standard could suggest some ‘blessing’ of a standard, regarding development of other standard
        • Current Phrasing: ‘Currently accepted standards’, ‘Choose among the competing best practices’
Short Version of the Code

• The principles summarize aspirations at a high level of abstraction
  – Without aspirations the details can become legalistic
  – The long version has clauses that give examples and details on how aspirations change the way software engineers act.

• ‘In accordance with their commitment to the health, safety and welfare of the public, software engineers shall adhere to Eight Principles.’

Eight Principles of the Code

1. PUBLIC - Software engineers shall act consistently with the public interest.

2. CLIENT AND EMPLOYER - Software engineers shall act in a manner that is in the best interests of their client and employer, consistent with the public interest.

3. PRODUCT - Software engineers shall ensure that their products and related modifications meet the highest professional standards possible.

4. JUDGMENT - Software engineers shall maintain integrity and independence in their professional judgment.

5. MANAGEMENT - Software engineering managers and leaders shall subscribe to and promote an ethical approach to the management of software development and maintenance.

6. PROFESSION - Software engineers shall advance the integrity and reputation of the profession consistent with the public interest.

7. COLLEAGUES - Software engineers shall be fair to and supportive of their colleagues.

8. SELF - Software engineers shall participate in lifelong learning regarding the practice of their profession and shall promote an ethical approach to the practice of the profession.
Eight Principles of the Code

1. PUBLIC - Software engineers shall act consistently with the public interest.

2. CLIENT AND EMPLOYER - Software engineers shall act in a manner that is in the best interests of their client and employer, consistent with the public interest.

3. PRODUCT - Software engineers shall ensure that their products and related modifications meet the highest professional standards possible.

4. JUDGMENT - Software engineers shall maintain integrity and independence in their professional judgment.

5. MANAGEMENT - Software engineering managers and leaders shall subscribe to and promote an ethical approach to the management of software development and maintenance.

6. PROFESSION - Software engineers shall advance the integrity and reputation of the profession consistent with the public interest.

7. COLLEAGUES - Software engineers shall be fair to and supportive of their colleagues.

8. SELF - Software engineers shall participate in lifelong learning regarding the practice of their profession and shall promote an ethical approach to the practice of the profession.
Web Sites

- IEEE-Computer Society/ACM
- The Software Engineering Ethics Research Institute
  - http://seeri.etsu.edu/
  - http://seeri.etsu.edu/Codes/TheSECode.htm (Code)
- ACM
  - http://www.acm.org/constitution/code.html
- CIPS
  - http://www.cips.ca/membership/ethics.htm
- American Society for Quality (ASQ)
  - http://www.asq.org/join/about/ethics.htm
- Quality Assurance Institute (QAI)