

Is there a “software engineering ethics”? Comparing commercial and research software engineering

Graham Lee

Department of Computer Science
University of Oxford
Oxford, United Kingdom OX1 3QD
Email: grahamlee@acm.org

Abstract—While ethical principles like fairness and honesty may be broad cultural or even universal virtues, many inputs into a professional software engineer’s ethical code will be situational, depending on the nature of the engineer’s employing institution and the line of business in which that organisation participates. We compare published codes of ethics for employees at Silicon Valley technology companies and at UK universities, to identify the commonalities and differences of the ethical values that guide engineering practice in each category of employer. This comparison represents the beginning of a context-dependent methodology for software engineering, in which practices are recommended based on the distinct properties of the problem domain and the solutions expected therein.

I. INTRODUCTION

Software engineering is a process in which one group of people create a software product that will be used by another, potentially different, group of people. Ethical considerations are important in how the software engineers work with each other, in the product that they create, and in the interactions mediated by the software in production. Aydemir and Dalpiaz identified a method in which ethical awareness could be instilled in the software engineering process[1].

We hypothesise that the ethical principles considered important in software engineering work will depend on the context of the work, particularly on the organisation sponsoring the software engineering and the ethical imperatives considered important in that organisation. We investigate this hypothesis by comparing the published codes of ethics for two different categories of institution known to hire software engineers. We identify common ethical imperatives in these codes of ethics, and explore the distribution of commonality across the institutions.

Section II describes the background to this research. Section III describes the method we used to discover and interpret the ethical codes used in two different classes of organisation that hire software engineers: U.S. technology companies and U.K. universities. The ethical imperatives discovered in the search of U.S. technology company codes are described in section IV, and those discovered in the search of U.K. university codes are described in section V. The two sources of ethical imperative are compared in section VI. Limitations to this work are discussed in section VII. Finally our conclusions are documented in section VIII.

II. BACKGROUND

The existence of a single, universal theory of software engineering is an open question in software engineering research. Johnson and Ekstedt argue for both the possibility of, and the value of, a Unified Theory in the discipline[2]. Perry and Batory outline that the nature of a General Theory of Software Engineering would describe many components, with “connector” theories describing the relationships between those components[3]. Conversely, Vessey describes the role of the application domain in understanding software engineering methodology, distinguishing “strong” approaches to problem solving that are tailored to specific problems from “weak”, generic approaches[4]. Glass solicited a taxonomy of application-specific methodologies according to their strengths and weaknesses at solving particular issues and a map of different problem domains with their specific needs onto those methodologies.[5]

An important source of differences between software engineering domains of application are the ethical considerations relevant in each domain. Ethical codes guide practice by constraining and influencing practitioners in their choice of actions[6]. I interpret the sources of questions of ethics in the practices of a software engineer as being:

- 1) in the software engineer’s professional interactions as a software engineer;
- 2) in the software engineer’s professional interactions in the software’s intended domain;
- 3) in the software’s capacity to support ethical or unethical actions taken by its users in its intended domain; and
- 4) in biases in decisions or outcomes generated by the system itself, in the case of machine learning and artificial intelligence systems.

Existing codes of ethics for software engineers are frequently concerned with the first of these four ethical aspects. The code of ethics and professional conduct from the Association of Computing Machinery (ACM) “is designed to inspire and guide the ethical conduct of all computing professionals, including current and aspiring practitioners, instructors, students, influencers, and anyone who uses computing technology in an impactful way”[7]. It is a document designed to guide computing professionals in using their privileged membership of the profession, and the specialist knowledge and skills

associated with the profession, in the public interest. The same is true of two other codes of practice for professional software engineers: that of the IEEE computer society[8] and that of BCS, the Chartered Institute for IT[9].

A reasonable critique of these codes is that while being universal in language, they are far from universal in adoption: the ACM claims a membership of “nearly 100,000” students and professionals while the U.S. Bureau of Labor Statistics estimates that 200,000 professional computer programmers work in that country alone[10]. Therefore the institutional codes of ethics such as that of the ACM do not apply to the professional practice of many software engineers. To find the ethical rules that these engineers do follow in their work, we should look to the codes of conduct issued by their employers.

There is also a growing body of research in the fourth area of ethical concern, namely the ethics of biases and decisions in artificial intelligence algorithms. Martin and Makoundo point to the growing interest in regulation of AI from both the United States and European Union, and describe the need for intentionality in designing the ethically-relevant aspects of AI systems[11]. Gerlich and Liozu describe a framework for considering ethical and legal compliance concerns in the design of algorithmic pricing functions[12]. With “algorithmic” biases and failures drawing high-profile criticism, such as the recent case of algorithmic bias in awarding school grades in the U.K.[13], the question of where software engineers gain ethical awareness and training is important.

This training does not come from formal educational programmes in software engineering. In a 2016 survey of its participants, the popular software engineering Q & A site Stack Overflow found that only 43% of respondents working in software ($n = 40183$) had a Bachelor-level degree in “computer science or a related field”[14]. A high-profile M.Sc. programme in software engineering does not have ethics on its syllabus[15]. This redoubles motivation to ask the question whether ethical considerations are being promoted by the employing organisations in their codes of conduct for employees.

When software engineering is performed without due consideration for the ethical implications, problems arise that affect the software engineering organisation, its customers, or society at large whether the software is created for commercial or for research ends. The Google Photos machine learning-based characterisation of user photos tagged black people in photographs as “gorillas”[16]. Researchers at Facebook collaborated on a publication that demonstrated how the stories chosen by their algorithmic news feed can affect the emotional state of their users[17], which resulted in negative coverage both in the academic community and wider news media due to the way in which the study was performed without informed consent and the implied power the company demonstrated over its users[18].

III. METHOD

We selected two distinct domains in which software engineers operate; publicly-traded American technology companies and UK universities. These two sets of institutions

are distinct in geographic culture, in purpose, and in the ways in which they are funded and managed, which should provide a good opportunity to find qualitative differences in professional ethics, even though the number of variables involved is too great to find causal relationships for distinctions observed. In other words if there are going to be differences between the codes of ethics in different organisations at all, we should be able to find evidence of them in the most different organisations. We then looked for public documents describing employee codes of ethics or conduct from these organisations. This was performed by searching for “[name of organisation] code of ethics” on the DuckDuckGo search engine.¹

Codes of ethics were found for five organisations in the American technology companies category: Amazon Inc.², Microsoft Inc.³, Google Inc.⁴, Netflix Inc.⁵ and Apple Inc.⁶. Codes of ethics were found for five organisations in the U.K. universities category: the Universities of Oxford⁷, Cambridge⁸, Warwick⁹, Bristol¹⁰ and Southampton¹¹.

Ethical imperatives active in the organisations were discovered by investigating the codes of conduct for statements recommending particular actions (“employees must” or “employees should”, for example) and proscribing other actions (“employees must not” or “employees should never”, for example).

We did not include imperatives that were stated to apply to non-software engineering roles, for example “the company’s accounting group must ensure that records are accurate”. This choice was made on the basis that the organisations under consideration are large enough that software engineering and accounting roles are probably filled by different people, and that these imperatives do not impact the day-to-day work of software engineers. Where a statement was ambiguous about coverage, for example “The Legal Department will designate certain employees who, based on their level of responsibility or the nature of their work, will be required to certify periodically that they have read, understand and complied with the Code of Conduct.”, we chose to include the statement as an ethical

¹DuckDuckGo, available at <https://duckduckgo.com>, is a public internet search engine similar to Google or Bing. The reason for choosing this search engine in particular is the author’s preference for their privacy policy.

²<https://ir.aboutamazon.com/corporate-governance/documents-and-charters/code-of-business-conduct-and-ethics/default.aspx>

³<https://www.microsoft.com/en-us/legal/compliance/sbc/download>

⁴Google’s Code of Ethics comprises the code for its parent company, Alphabet Inc. at <https://abc.xyz/investor/other/code-of-conduct/> and one specific to Google at <https://abc.xyz/investor/other/google-code-of-conduct/>. They each explicitly refer to the other so are taken as a single code here.

⁵https://s22.q4cdn.com/959853165/files/doc_downloads/governance_docs/73.pdf

⁶https://s2.q4cdn.com/470004039/files/doc_downloads/Business-Conduct-Policy.pdf

⁷<https://hr.admin.ox.ac.uk/files/airupdate2020pdf>

⁸<https://www.research-integrity.admin.cam.ac.uk/research-integrity/research-integrity-statement>

⁹https://warwick.ac.uk/services/ris/research_integrity/code_of_practice_and_policies/research_code_of_practice

¹⁰http://www.bristol.ac.uk/media-library/sites/red/documents/research-governance/Ethics_Policy_v8_03-07-19.pdf

¹¹<https://www.southampton.ac.uk/about/governance/policies/ethics.page>

imperative on the basis that a software engineer could be in the category of designated employees.

We also discounted imperatives that employees consult an internal policy document or familiarise themselves with other institutional guidelines, on the basis that these other documents were not universally available. We note in section VII the limitations caused by this decision. Conversely, we recorded imperatives that ask employees to refer to *external* documents such as professional codes of practice or relevant laws, as this indicates an awareness of the social situation of work performed within the organisations.

The discovered imperatives were then coded and tabulated, facilitating comparison between multiple organisations. Some sentences in these documents capture aspirations or values but not in an imperative sense; for example “All of our communications and other interactions with our users should increase their trust in us.” This sentence expresses a desired outcome, but does not represent a specific expectation that certain behaviour is adopted or proscribed. Such sentences were not coded as ethical imperatives.

A total of 143 different ethical imperatives were recorded across the documents investigated. The resulting table of ethical imperatives, and the codes in which those imperatives appeared on the dates they were accessed, is available at <http://doi.org/10.5281/zenodo.4472897>. An indication of the imperatives identified and their prevalence across the different codes of practice is given in table I.

IV. COMMERCIAL SOFTWARE ENGINEERING ETHICS

The five corporate codes of conduct from U.S. publicly-traded technology companies varied significantly in scope, with the shortest providing nine identifiable imperatives and the longest 67. All five agree on four imperatives: that employees should avoid conflicts of interest, should declare conflicts of interest where they arise, should follow a recommended procedure for reporting ethical misconduct and should not retaliate against colleagues who report misconduct.

Other areas of broad agreement relate to the fiduciary responsibilities of the selected organisations as publicly-traded companies: four out of five codes require employees to follow all relevant laws; comply with insider trading restrictions; refrain from illegal discrimination or harassment, acting in a violent or threatening manner, or working under the influence of illegal drugs; and avoid bribery. The final items on which four of the five codes of conduct agree are related to guidance and support in following the code: employees should raise legal questions with their Legal department; should speak with managers for guidance in following the code; and can use anonymous tools for reporting violations if appropriate.

There were 4 imperatives on which all five codes agreed; 10 that were contained by four of the codes; 12 that were in three codes; 28 imperatives in two of the codes; and 58 imperatives that each appeared in a single code studied.

Perhaps surprisingly, despite all of the organisations in this section being computing companies, very little mention is made in their corporate codes of practice of the software they

create and its impact on society. Two of the codes tell employees that they should protect user data from unauthorized use, and respect their users’ freedoms of expression and privacy. Other imperatives about the software produced or the software engineering process at these companies are only mentioned in one code each. The Alphabet/Google code of practice tells employees to report any time users aren’t being well-served, and to fix anything they see that is “broken”. Microsoft’s code tells employees to provide clear and accurate privacy notices; to use customer data only in ways the customers agreed to; to make accessible products and to provide clear accessibility information. Apple’s code tells employees to seek approval before working on technical standards or integrating open-source software.

The conclusion we draw from this observation is that the codes of conduct for these large corporations do not demonstrate their executives’ commitment to using the special computing knowledge of their professional employees for the public good. The codes of ethics demonstrate that the employees have been told not to violate relevant corporate law and Securities and Exchange Commission rules.

V. RESEARCH SOFTWARE ENGINEERING ETHICS

The code of ethics produced by U.K. universities are typically documents about the integrity of research produced at the universities. As with the commercial codes, they varied in length, with the longest containing 24 imperatives, and the shortest 10 imperatives. All of the examined codes agree that researchers should be aware of the codes of ethics relevant to their domain, and those of professional societies in those domains. Four of the five require researchers to be honest in their research; to ensure the accuracy of results and data; to acknowledge the contributions of others; to declare conflicts of interest; and to comply with statutory ethical obligations.

Despite the high importance placed on writing software to the production of research by many scientists[19][20], there were no imperatives identified in the academic codes of practice that referred to the ethics of research software.

There were two imperatives present in all five codes; four that appeared in four of the codes; eight in three of the codes; four in two; and 24 imperatives that only appear in one academic code studied.

VI. COMPARISON OF ETHICAL CODES IN THE TWO DOMAINS

Only four of the ethical imperatives identified in this study were present in at least six of the codes of ethics, meaning that a majority of the institutions recognised the imperatives across both categories of organisation. These were the imperatives to avoid conflicts of interest; to declare conflicts of interest; to follow guidelines for reporting misconduct; and to refrain from retaliation against those who reported misconduct.

Only nine of the 143 different ethical imperatives identified appeared in at least one of the U.S. corporate codes of conduct and at least one of the U.K. university codes of research integrity. The codes of ethical conduct were found

Imperative	Ccommercial codes	Academic codes	Total occurrences
Act in the best interest of the organisation	1	0	1
Follow all relevant laws	4	0	4
Avoid Conflicts of Interest	5	4	9
Obtain approval before working on technical standards	1	0	1
Bring dogs to the workplace in conformance to policy	1	0	1
Aim to design, produce and disseminate work of the highest quality and ethical standards.	0	1	1

TABLE I
ILLUSTRATIVE EXAMPLES OF ETHICAL IMPERATIVES DISCOVERED.

in general to be heavily influenced by the domain of the sponsoring organisation. Codes of practice from U.S. publicly-traded technology firms focus on the fiduciary responsibilities of the directors, with guidelines designed to prevent insider trading and anti-competitive behaviour but little to no acknowledgement of the ethical impact of these company's products. Codes of practice from U.K. universities concentrate on the openness, accuracy, and integrity of the academic record and their employees' contribution to that record. We therefore conclude that the information gained in this study supports our hypothesis that the ethics relevant to software engineering practice is dependent on the context in which that engineering is performed.

VII. LIMITATIONS AND THREATS TO VALIDITY

This study was performed with only one person—the author—interpreting the codes of practice and deciding whether imperatives worded differently in the various documents had the same meaning. A common practice is to have multiple researchers independently code the statements and compare results for consistency, which would remove bias from the coding and improves the quality of the study.

This study relied on the public sharing of codes of ethics by the organisations investigated. Software engineers in these organisations may have other sources of ethical imperatives, such as internal documents shared on corporate intranets or normative behaviours transmitted culturally within their teams. These other sources of ethical values would not be discovered by this survey. Some organisations were considered but excluded from the study because a published code of ethics could not be identified: this included American technology companies that do not publish a code of ethics, U.K. universities that delegate ethics policies to faculties or departments, and U.K. universities that refer matters on ethics policy to external documents such as the Universities UK Concordat for Research Integrity[21].

The extent to which different organisations placed ethical imperatives in their published codes of practice documents varies, for example the U.K. universities differ on whether the ethical practices of research involving humans, animals, or human tissue are mentioned in their main research codes or in ancillary documents. This could reflect the importance of those fields to the different institutions, but regardless may mean that comparing stated codes of conduct is not a like-for-like comparison.

VIII. CONCLUSIONS AND FURTHER WORK

There is not much overlap between the codes of ethics found in U.S. technology companies, and in U.K. universities. This suggests that there are different values desired in the work produced in each type of organisation, but rather than indicating different ethical contexts for software engineering in the different domains, the reason is that the ethics policies at the institutions studied do not give much (if any) consideration to the performance of software engineering in those institutions, despite the centrality of software to their respective enterprises. If software engineers are cognisant of ethical implications to their work, this awareness is not central to the organisations they work in.

Despite the largely non-overlapping ethical statements that were discovered, it is common to find the same software engineering methodologies used in the two domains: for example Agile software development which arose in the American commercial context is also used by academic software engineering projects[22]. This could be taken as an indication that the performance of software engineering is not typically troubled by ethical concerns: that there is no need to adapt processes or methodologies to the ethical context of the sponsoring organisation.

Another interpretation is that it represents a limitation in the current practice of research software engineering. If research integrity is a significant factor in the production of research, then perhaps we should expect different methodologies to be used in creating the software that supports and enables the research. If this latter is the case, then the answer to the question “is there a software engineering ethics?” is not “no”; it is “there are many”.

This question could be explored by “triangulating” this quantitative work with a qualitative analysis of how the ethical imperatives discovered in this study affect the day-to-day work of software engineers within the considered organisations[23]. Interviews of software engineers in these organisations could be conducted to uncover how important the institutional codes of conduct are to their work, and other sources of ethical context, and workplace ethnography studies could elucidate the frequency and manner in which ethical issues must be navigated by working software engineers.

There is value in surveying software engineers to discover what training they receive from their employing institutions in ethics, and whether that ethical training incorporates concepts from the institutions' codes of conduct or other sources. This work would help to answer questions about how the ethical

context of a software project manifests in the decisions made and tasks performed in completing the project.

Future work could extend the survey here to more institutions, to uncover statistical differences in the prevalence of ethical imperatives in the different categories of institutions and to uncover correlations between the ethical codes and other variables such as the type or location of the institution.

It would be valuable in particular to compare the ethical frameworks of research software engineers working in academic and commercial institutions: for example, research software engineers in a university and engineers in a technology company's research department. Similarly, the ethics of commercial software engineers in commercial organisations could be compared with those of academic software engineers working in business schools or on spin-out company projects.

The prevalence of common methodologies such as Agile software development in many contexts despite the large observed variation in ethical values relevant to these contexts raises the question of an "ethics gap" in software engineering methodology. Does current best practice in software engineering appropriately surface and address ethical issues, and how could modifications to the processes improve the ethical compliance of software engineering practice?

IX. DATA AVAILABILITY

All data used in this paper are available at <http://doi.org/10.5281/zenodo.4472897> and licensed under the terms of the Creative Commons CC BY-NC-ND 4.0¹² licence.

X. DECLARATION OF INTERESTS

The author has connections to two of the organisations studied in this work: he is currently an employee of the University of Oxford, and has experience as a professional software engineer including a period contracting for Apple, Inc. The author has no material interests or investments in any of the organisations studied.

REFERENCES

- [1] F. B. Aydemir and F. Dalpiaz, "A Roadmap for Ethics-Aware Software Engineering," in *2018 IEEE/ACM International Workshop on Software Fairness (FairWare)*, 2018, pp. 15–21.
- [2] P. Johnson and M. Ekstedt, "In Search of a Unified Theory of Software Engineering," in *International Conference on Software Engineering Advances (ICSEA 2007)*, 2007, pp. 1–1.
- [3] D. E. Perry and D. Batory, "A Theory about the Structure of GTSEs," in *2015 IEEE/ACM 4th SEMAT Workshop on a General Theory of Software Engineering*, 2015, pp. 39–46.
- [4] I. Vessey, "Problems versus solutions: The role of the application domain in software," in *Papers Presented at the Seventh Workshop on Empirical Studies of Programmers - ESP '97*. Alexandria, Virginia, United States: ACM Press, 1997, pp. 233–240.
- [5] R. L. Glass, "Matching Methodology to Problem Domain," *Commun. ACM*, vol. 47, no. 5, pp. 19–21, May 2004.
- [6] M. Weinberg, "A Case for an Expanded Framework of Ethics in Practice," *Ethics & Behavior*, vol. 15, no. 4, pp. 327–338, Oct. 2005.
- [7] A. C. . T. Force, "ACM Code of Ethics and Professional Conduct," <https://www.acm.org/code-of-ethics>, Jun. 2018.
- [8] IEEE-CS/ACM Joint Task Force on Software Engineering Ethics and Professional Practices, "IEEE Computer Society Code of Ethics," <https://www.computer.org/education/code-of-ethics>, 1999.

- [9] Trustee Board of the British Computer Society, "BCS Code of Conduct," <https://www.bcs.org/membership/become-a-member/bcs-code-of-conduct/>, Jun. 2015.
- [10] U.S. Bureau of Labor Statistics, "Occupational Employment and Wages, May 2019," <https://www.bls.gov/oes/current/oes151251.htm>, May 2019.
- [11] C. D. Martin and T. T. Makoundou, "Taking the High Road: Ethics by Design in AI," *ACM Inroads*, vol. 8, no. 4, pp. 35–37, Oct. 2017.
- [12] J. A. Gerlich and S. M. Liozu, "A Conceptual Framework of Ethical Considerations and Legal Constraints in the Algorithm-Driven Pricing Function," in *Proceedings of the Ninth International Conference on Engaged Management Scholarship*, Sep. 2019.
- [13] S. Coughlan, "Why did the A-level algorithm say no?" *BBC News*, Aug. 2020.
- [14] S. Overflow, "Stack Overflow Developer Survey Results 2016," <https://insights.stackoverflow.com/survey/2016#developer-profile-education>, 2016.
- [15] University of Oxford Software Engineering Programme, "University of Oxford Courses in Software Engineering—Subjects," <http://www.cs.ox.ac.uk/softeng/courses/subjects.html>.
- [16] A. Griffin, "Google Photos Tags Black People as 'Gorillas', Puts Pictures in Special Folder," *The Independent*, Jul. 2015.
- [17] A. D. I. Kramer, J. E. Guillory, and J. T. Hancock, "Experimental evidence of massive-scale emotional contagion through social networks," *Proceedings of the National Academy of Sciences*, vol. 111, no. 24, pp. 8788–8790, Jun. 2014.
- [18] K. Hill, "Facebook Manipulated 689,003 Users' Emotions For Science," *Forbes*, Jun. 2014.
- [19] J. E. Hannay, C. MacLeod, J. Singer, H. P. Langtangen, D. Pfahl, and G. Wilson, "How do scientists develop and use scientific software?" in *2009 ICSE Workshop on Software Engineering for Computational Science and Engineering*, May 2009, p. nil.
- [20] G. Pinto, I. Wiese, and L. F. Dias, "How do scientists develop scientific software? An external replication," in *2018 IEEE 25th International Conference on Software Analysis, Evolution and Reengineering (SANER)*, 2018, pp. 582–591.
- [21] U. UK, "The concordat for Research Integrity," <https://www.universitiesuk.ac.uk/policy-and-analysis/reports/Pages/the-concordat-for-research-integrity.aspx>, Oct. 2019.
- [22] M. T. Sletholt, J. Hannay, D. Pfahl, H. C. Benestad, and H. P. Langtangen, "A literature review of agile practices and their effects in scientific software development," in *Proceedings of the 4th International Workshop on Software Engineering for Computational Science and Engineering*. ACM, May 2011, pp. 1–9.
- [23] M. Petre, J. Buckley, L. Church, M.-A. Storey, and T. Zimmermann, "Behavioral Science of Software Engineering," *IEEE Software*, vol. 37, no. 6, pp. 21–25, 2020.

¹²<https://www.creativecommons.org/licenses/by-nc-nd/4.0/>