Ethical Issues in Software Engineering Research: A Survey of Current Practice

TRACY HALL   t.hall@herts.ac.uk
Department of Computer Science, University of Hertfordshire, Hatfield, AL10 9AB, UK

VALERIE FLYNN   flynnv@sbu.ac.uk
School of Computing, South Bank University, London, SE1 0AA, UK

Abstract. In this paper we explore how the software engineering research community is currently dealing with the ethical issues that some empirical research presents. We discuss how the immaturity of the software engineering discipline is reflected in an approach to ethical issues that compares unfavourably with other more mature disciplines.

We show that an analysis of recent published work measures an increase in empirical software engineering research currently being undertaken. We also discuss our survey of UK University Department Heads which explores how the software engineering research community is dealing with the ethical issues related to empirical work. Overall we found that whilst some UK Universities have taken ethical issues very seriously, others have not considered the issues.

Keywords: Software engineering, ethics, empirical research, survey

1. Introduction

In this paper we present survey results collected from 44 Computer Science Departments in UK Universities. We report detailed questionnaire findings describing how the ethical considerations of software engineering research are dealt within the UK. We use this data to suggest ways that ethical considerations can be established in software engineering research involving human subjects.

There has been a significant increase in software engineering research involving the use of human participants in recent years. We reviewed IEEE Transactions on Software Engineering (TSE) over the period 1985–1999 to reveal a year-on-year increase in publications that use human participants (see Appendix 1). We found that in the period 1985–1990, for example, the number of articles describing research involving human participants was only 8, compared with 19 in the period 1991–1996 and 23 in the period 1997–1999. As TSE is one of the most technically oriented software publications, this growth in research probably underestimates the true extent of the research being undertaken. Indeed, Seaman and Basili (1998) and others argue that the people component of software engineering is now fundamental to the discipline.

Although the use of human participants in software engineering research seems to be becoming more prevalent, little attention has been given to the ethical issues this may raise. In medicine, law, psychology and the more traditional disciplines, the

ethical problems involved in using humans has long been recognised and a regulatory framework has been established to protect participants from possible harm (Beach, 1996; Kimmel, 1996). This trend has gone furthest in the US, where an act was passed as early as 1974 (National Research Act) to protect human subjects of biomedical and behavioural research (Kimmel, 1996). No such law exists in the UK: however, fear of litigation arising from contravention of the 1998 Data Protection Act and other legislation designed to protect human rights has provided a strong incentive to many institutions to put a regulatory framework in place.

The research involving human participants reported in TSE indicates that some researchers are aware of the ethical issues raised by their research. There are certainly instances of researchers explaining their procedures for gaining voluntary consent and debriefing participants (Griswold et al., 1998; Jarke et al., 1985; Murphy et al., 1999). However there is an over-reliance on the use of student participants and little explicit mention of the steps taken to guard their rights.

The moral and legal consequences of misusing human participants means it is essential software engineering research maintain the highest ethical standards. There is little data, however, on the attitudes of the software engineering community towards ethical issues or the steps taken to monitor and regulate the conduct of software engineering research involving human participants. We could find no other work reporting on how the ethical dimension of empirical software engineering research is currently being handled. The research we describe in this paper goes some way towards filling this gap. We present a comprehensive snapshot of the UK state-of-practice regarding ethical issues in software engineering research. Our findings confirm that there is a generally *ad hoc* approach to ethical issues in UK software engineering research. We present our findings as a baseline from which the ethical aspects of research can be improved.

In Section 2 of this paper we describe our research methods. In Section 3 we present our main findings. We discuss the implications of our findings in Section 4 and draw some conclusions in Section 5. In Section 6 we provide an overview of ways in which the ethical dimension of software engineering research could be strengthened.

2. Research Methods

In this work we explore how the UK research community is dealing with the ethical considerations that the growth in software engineering research involving human participants implies. Our aim was to see how UK University Departments are formally considering the ethical dimensions of some of the research that is being done. Over a 3 month period from July to September 2000 we conducted a national survey of practice in the UK research community.

We sent detailed questionnaires to all 94 University Computing Department Heads in the UK. (The questionnaire we used can be viewed at http://homepages. feis.herts.ac.uk/~pppgroup/EthicsQuestionnaire.htm.)

The survey was divided into six main sections. The first was designed to elicit information on the size of the respondents' department, the number of software engineering academics/researchers employed and the amount of software engineering research which was being undertaken by the department. This contextual data was necessary to establish whether there were any links between the size of the department, the scale of software engineering research activity and importance ascribed to ethics.

We posted questionnaires directly to Heads of Departments. We recorded a questionnaire response rate of 47% with 44 heads of department returning a completed questionnaire. We used three reminders via post and e-mail to achieve this response rate. For a postal survey of this nature this response rate is considered excellent (Berdie and Anderson, 1974).

3. Research Findings

3.1. Software Engineering Research: Size and Effort

Table 1 gives an indication of the size of computing departments in the UK. It also shows that software engineering forms a relatively small component of the overall work of computing departments. Furthermore Table 1 also shows that software engineering research activity remains relatively low in many departments with 64% of Departments having less than 10 people doing software engineering research of any sort.

Table 2 presents a breakdown of the type of software engineering research that UK departments undertake. It shows that almost half of the departments put most of their research effort into theoretical and systems development orientated research. It also shows that about a quarter of departments do some empirical research, with only two doing a great deal of empirical research, however most departments that do any software engineering research do some empirically based work.

Table 1. Effort in software engineering research.

Academic staff numbers	Whole computing department		Software engineering component		Active software engineering researchers	
1–10	0	0	20	45%	28	64%
11–25	11	25%	15	34%	12	27%
26–50	18	41%	8	18%	4	9%
51–75	9	20%	0	0	0	0
76–100	3	7%	0	0	0	0
Over 100	3	7%	0	0	0	0
Total	44	100%	44	100%	44	100%

Table 2. Types of software engineering research.

	Type of research undertaken	Overall software engineering research effort in computing departments	
		1–50% of total effort	51–100% of total effort
Non-empirical	Systems development	17 (39%)	13 (30%)
	Theoretical work	23 (52%)	5 (11%)
Potentially empirical	Case study-based	10 (28%)	2 (5%)
	Questionnaire-based	13 (30%)	0
	Interview-based	11 (25%)	0
	Observation-based	12 (27%)	1 (2%)

NB Many Heads declined to reveal the nature of the software engineering research being undertaken.

3.2. Importance of the Ethical Dimension

Table 3 shows that only 16 of the 44 Heads believed that monitoring the ethical considerations of empirical software engineering research is very important. This is a surprisingly low number considering we were asking their hypothetical opinion. However it probably reflects the relatively few departments which actually undertake

Table 3. Importance of ethical consideration.

How important do you think monitoring the ethical considerations of empirical research is?	
Very important	16 (36%)
No feelings either way	17 (39%)
Not important	8 (18%)
Don't know	3 (7%)
Total	44 (100%)

Table 4. Effectiveness of current measures.

How well do you think your University monitors the ethics of its software engineering research?	
Very effectively	1 (2%)
Reasonably effectively	10 (24%)
No feelings either way	12 (29%)
Not very effectively	11 (26%)
Not effectively at all	4 (10%)
Don't know	(10%)
Total	42 (100%)

very much empirical work (as shown in Table 2). Indeed closer analysis of responses showed that the Heads who believed that the ethical dimension was important, were those where most empirical work was being undertaken. However comments annotated to some questionnaires showed alarming animosity to, first, doing empirical software engineering research and, second, to the idea that the ethical aspects of research in software engineering should be formally considered.

Table 4 reveals mixed views about how effectively universities are currently monitoring software engineering ethnics.

3.3. Ethical Mechanisms

Table 5 shows that of the 44 Department Heads who responded, 11 required that the ethical aspects of software engineering research projects were formally approved by the university. Given the importance of ethical issues within certain empirical software engineering projects, we were concerned that relatively few Universities in the UK seem to be monitoring the issues at all. However, this result may, again, be related to the results presented in Table 2 that suggest that empirical research in software engineering is still nascent in many UK Universities and so there is not yet a strong imperative to monitor the issue in the UK.

We noted the following comments from some respondents regarding formal consideration of ethics:

> "I find this questionnaire very worrying because the idea of having to seek ethical approval threatens academic freedom."

> "(Seeking ethical approval) has never arisen and I don't know why this is an issue" (respondent did not answer the question regarding the type of software engineering research his/her department undertakes.)

> "Although we are required to seek ethical approval to my knowledge it has never happened!"

> "Procedures have been designed for Health Professionals and Psychologists, so it's difficult to persuade computing people that it is sensible for them."

> "Maybe a sledge hammer to crack a nut."

Table 5. Universities with procedures to oversee ethical issues in software engineering research.

	Yes	No	Don't know
Does your University require any SE research to seek ethical approval?	11 (25%)	26 (59%)	7 (16%)

Furthermore, several respondents noted comments similar to the following:

"It is up to the individual researcher to do the right thing."

"Each individual takes responsibility for ethical considerations."

"No one is responsible for the ethical approval of CS research."

Table 6 summarises the monitoring mechanisms in place for overseeing the ethical aspects of empirical research in the 11 Universities with a formal ethics process. It shows that for the universities with approval mechanisms:

– There are an array of communication mechanisms in place, though university regulation documents are relied most on to impart information about ethics.

– University committee is the most popular way of monitoring ethical aspects of projects.

– Although all 11 Universities require postgraduates and academic staff to seek ethical approval, only half expect undergraduates to do the same.

All 11 Department Heads reported that sanctions are in place where ethics procedures are not followed. However, nine of the 11 did not know what sanctions are possible and only two respondents said that researchers could be officially reprimanded about ethical issues.

3.4. Rate of Approval

When we asked Heads about the actual ethical approval processes that were in place, eight of the 11 Universities with a process in place report an ethics approval application rate of between one and 10 applications per year for software engineering projects. Six of the 11 said that this rate has generally remained steady, with another five saying the rate has been steadily increasing over the years. Although we need to

Table 6. Monitoring mechanisms in place.

Communication of ethical approval requirements ($n = 11$)		Ethics monitoring mechanisms ($n = 11$)		Ethics approval criteria ($n = 11$)	
University regulations documents	11 (100%)	Departmental or faculty committee	5 (45%)	Undergraduate students	6 (54%)
Line managers	7 (64%)	University committee	9 (82%)	Postgraduate students	11 (100%)
Training	7 (64%)	Head of Department	3 (27%)	Academic staff	11 (100%)

know more about the details of the projects applying before we can comment in detail, it is, again, interesting to think about why the application rate is so low considering the increased rate of such work we report earlier.

We also found that the time applications take to process varies between Universities. Four Department Heads say that applications generally take less than a month to process with another four taking between 1 and 3 months on average. The final three Department Heads did not know how long the process took.

Respondents also told us that most applications were approved with either no changes or only minor changes. Only two Department Heads could recall any applications being rejected.

When we asked Heads for their feelings on whether all research that should have ethical approval actually has it, only one of the 11 said that they are highly confident that all software engineering research that should have ethical approval has actually applied for it. Four Department Heads say that they are not very confident and the remaining six say they are reasonably confident.

3.5. Training in Key Ethical Issues

Table 7 shows how much training in ethical issues are offered to software engineering researchers. It shows that very little training is provided nationally in the UK to Ph.D. students. Further analysis of these results showed a relationship between whether an institution requires ethical approval and the provision of training in ethics to researchers. Departments with approval processes in place are much more likely to offer guidance on the ethics of research to Ph.D. students.

We wanted to find out what ethical issues software engineering researchers were taught. Respondents were asked to indicate whether they offered training in data protection legislation, risks of using human participants, moral theories, the politics of field research, ethics of emerging technologies etc. The results indicate that while many provide guidance in legislative matters, other issues are often neglected. It is particularly worrying that only 20% of respondent Universities provided guidance on cross-cultural differences in ethics and 19% on the ethics of emerging technologies (not shown in tables).

Table 7. Groups trained in ethics.

Teaching ethics to various groups of students	($n = 44$)
Undergraduates	35 (79%)
Postgraduates	23 (52%)
Ph.D's	11 (25%)

Table 8. Type of training given.

Formal guidance to researchers in various issues	(n = 44)
Data protection legislation	26 (57%)
The use of human subjects in research	13 (30)
Publication ethics	18 (41%)

4. Discussion

4.1. Size of Research Effort

Our results reveal a surprisingly small amount of software engineering research in the surveyed Universities. Furthermore our results also suggest that a very small amount of that research is empirically based work. Indeed if our result are representative for the whole UK sector, it means there is a surprisingly small amount of empirical research being undertaken in the UK. This appears to conflict with what we reported earlier, where we show an increase in empirical software engineering research published in TSE. Of course, our UK data does not reveal how software engineering research activity has changed over the years, as our TSE data does.

4.2. Approval Processes

There are a number of worrying issues that our results highlight related to the approval of research:

Responsibility. It is very worrying that a significant minority of Department Heads believe that responsibility for ethical research methods rests solely on the individual. Furthermore these Universities appear to have no monitoring processes in place. This attitude concerns us greatly and could actually conflict with what researchers' contracts of employment say and the actual responsibilities of the institution in law.

Inconsistency. In Universities with approval processes, it is also interesting to speculate on the reasons that Universities differentiate between postgraduate and undergraduate projects. Many Universities do not require undergraduates to seek approval. This inconsistency is difficult to explain given the number of undergraduate projects that are undertaken each year. Certainly in our own Universities it is common for students to use questionnaires during the system requirement capture phase of their project and to use human subjects to evaluate their finished systems. This would probably be an excellent opportunity to start educating potential future researchers about the importance of considering the ethical dimension of using human subjects in software engineering.

Timeliness. Our results also suggest that the length of some approval processes is problematic. We believe an approval process that can take 3 months introduces a significant dis-incentive to both doing empirical work and applying for ethical approval.

Structures. Our findings suggest that Universities that require ethical approval are also more likely to provide structures to help researchers conduct high quality research. This suggests that there is a relationship between a University's approach to requiring formal approval and provision of supporting structures. Such Universities are likely to produce researchers that are in a position to respond to ethical issues appropriately. This is an important finding as it also implies the converse, ie, that Universities which do not require formal consideration of ethics, also do not support researchers in other ways to account for ethical issues.

4.3. Training in Ethics

Our results on training researchers suggest that there are training gaps in University provision. These gaps are especially evident in considering issues such as risks of using human participants, moral theories, the politics of field research, ethics of emerging technologies etc. As research efforts become increasingly global and new technologies are introduced which may pose threats to society, software engineering researchers need to aware of cultural differences in ethical standards and beliefs and be aware of the moral implications of research with new technologies.

Our findings on training are very worrying considering the serious nature of inappropriately dealing with the ethical aspects of working with human subjects, and of the sanctions against researchers that Universities say they have in place. It seems that most Universities' training in this area is less than thorough. This further confirms the stated attitudes of some Heads when they said that they believed it was up to the individual researcher to be responsible for any ethical issues. It is likely, however that once the worldwide growth in empirical software engineering that we report from TSE, is reflected in the UK's software engineering research profile, problems will become apparent because of lack of training. Poor consideration of the ethical factors of empirical research is likely to result in official complaints from research participants and litigation is possible. Probably once this situation occurs training will be provided for researchers as an imperative.

5. Conclusions

The results clearly suggest that ethical issues involved in conducting research with human participants is accorded a fairly low priority by most Heads of Department. There are many reasons this may be the case. One may be the belief that ethical issues are a matter for the individual's own conscience; that it is not the responsibility of the institution to dictate software engineering researchers' morals. There is some

support for this in the Heads' comments about academic freedom and the right of individuals to make their own moral choices. This argument has some merit but it overlooks the rights of human participants in research, many of which are now enshrined in law. Another reason may be the relatively small scale of software engineering research activity. If so little is being carried out, Heads of Department and the institutions to which they belong may feel that ethical issues do not warrant attention or not wish to discourage the efforts of researchers. This is understandable but failure to recognise the rights of human participants in software engineering research brings into question the value of the research being undertaken.

A third reason for the apparent complacency may be the culture of technical departments. If most of the staff working in computing departments come from a technical background, it is possible that they may not have been taught research ethics and/or may see no need for it in domains which have traditionally been dominated by technical concerns. Given that software engineering and computing research generally is becoming more people-centred, this bias against empirical research in computing research is surprising.

A final reason for the heads' complacency may be the culture of the institutions. Our findings suggest that the institutions are often slow in responding to requests for ethical approval and usually return them unaltered. While the latter could be regarded as evidence that software engineering research is highly ethical, it may also be that the committees are not as rigorous as they might be or that there is a perception that software engineering has little to do with people and therefore cannot pose any ethical dilemmas. Without more detailed scrutiny of the vetting procedures, it is difficult to tell. Some combination of all the above factors probably accounts for the complacency revealed by the results of our survey.

6. Possible Ways Forward

The lack of awareness of, or concern about, ethical issues revealed in the survey could be addressed in a number of ways. Firstly, the software engineering community itself needs to be encouraged to take ethical issues more seriously. The publication of the Special Issue is a significant step in this direction but other measures are needed. It would be helpful, for example, if there were a Code of Conduct to guide researchers. At present, software engineers in the UK are covered by the British Computer Society (BCS) and the Institute of Electrical Engineers (IEE) codes of conduct. While excellent in many respects, they do not deal explicitly with the use of human participants in empirical research. Given the anticipated growth of studies in this area, perhaps it is time either for the codes to be updated or for the Software Engineering community to develop its own code. Such a code, based on experience of conducting empirical studies within software engineering, would underline the importance attached to ethical issues by the software engineering community.

The profile of research ethics within the software engineering community could also be enhanced by encouraging conference organisers to solicit papers that deal with research ethics. This may be difficult because some organisers may feel that ethics is not an important or interesting area to cover. Careful presentation of material and selection of papers can help to overcome this objection. Papers should focus on issues that are relevant to software engineering researchers and highlight controversial cases or instances of abuse from which lessons can be drawn. Even if only a few papers appear in conference proceedings, it will be a sign that ethical issues are beginning to receive serious attention.

Changes in the software engineering community's perception of ethical issues may help to change attitudes amongst researchers but this is likely to be a slow process. Our research suggests that change needs to take place urgently within the Universities themselves. The apparent laxity of University procedures for regulating software engineering research probably reflects the low importance ascribed to research ethics in general. This is a difficult problem to resolve because it relates to the Universities' value systems and the legal framework within which they operate. Unlike the US, the UK does not have a strong regulatory system for dealing with scientific misconduct (Hagmann, 1999).

One of the ways good practice could be enhanced is through provision of training in research ethics. Our research indicates that few institutions offer such training, particularly at postgraduate levels. External agencies that validate courses and funding bodies could perhaps be encouraged to look more closely for evidence that ethical issues are being taught. This already happens to some extent. When the BCS validates professional courses it usually seeks assurance that professional issues are covered. This could be enlarged to cover areas associated with research projects. The funding bodies could likewise look for evidence that ethical issues are being addressed.

Important though it is to teach students about research ethics, this will not automatically lead to improvements in ethical behaviour. To have any effect, the taught material must describe the circumstances in which malpractice is likely to occur. More research in software engineering is needed to establish these circumstances and provide guidelines on good practice. Those teaching students also need to set a good example and create a climate in which ethical conduct is expected. The importance of this was recently highlighted at a conference on ethics in biomedical research.[1] Those teaching software engineering in our Universities therefore have a key role to play in promoting awareness of research ethics amongst students and in the institutions in which they work.

Notes

1. Ringberg Conference on Ethics in Research, 20–23 Oct. and Joint Consensus Conference on Misconduct in Biomedica; Research, Edinburgh 28–29 Oct. 1999.

Appendix 1

Empirical work published in IEEE Transactions on Software Engineering 1985–1999
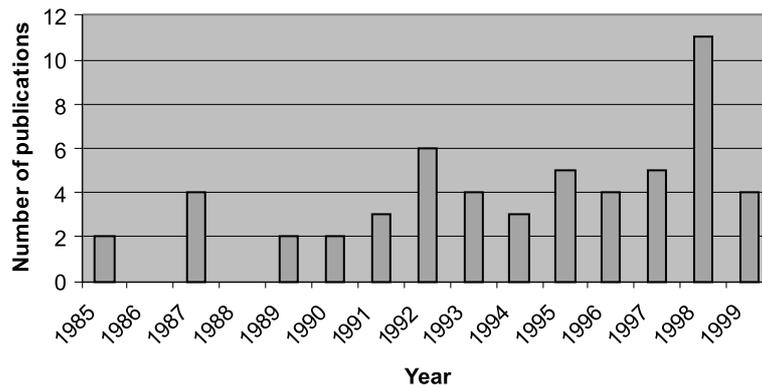
A summary of the published papers we analysed can be found at http://homepages. feis.herts.ac.uk/∼pppgroup/IEEEtse.htm

References

Beach, D. 1996. *The Responsible Conduct of Research*. VCH Verlagsgesellschaft mbH, D-69451.

Berdie, D. R., and Anderson J. F. 1974. *Questionnaires: Design and Use*, Pennsylvania: Scarecrow Press.

Griswold, W. G., Chen, M. I. Bowdidge, R. W., Cabaniss, J. L., Nguyen, V., and Morgenthaler J. D. 1998. Tool Support for Planning the Restructuring of Data Abstractions in Large Systems. *Transactions on Software Engineering* 24(7): 534–558.

Hagmann, M. 1999. Europe Stresses Prevention Rather than Cure. *Science*. 17th Dec. v286, issue 5448, p. 2258.

Jarke, M., Turner, J. A., Stor, E. A., Vassilou, Y., White, N. H., and Michielsen, K. 1985. A Field Evaluation of Natural Language for Data Retrieval. *IEEE Transactions on Software Engineering* 11(1): 97–114.

Kimmel, A. 1996. *Ethical Issues in Behavioral Research: a Survey*. Oxford, UK: Blackwell.

Murphy, G., Walker, R. J., and Baniassad, E. L. 1999. Evaluating Emerging Software Development Technologies: Lessons Learned from Assessing Aspect-Oriented Programming. *IEEE Transactions on Software Engineering* 25(4): 438–455.

Seaman, C. B., and Basili, V. R. 1998. Communication and Organization: An Empirical Study of Discussion in Inspection Meetings. *IEEE Transactions on Software Engineering* 24(7): 559–572.

Dr. Tracy Hall is a Principal Lecturer in software engineering at the University of Hertfordshire. She heads the Empirical Software Engineering Group in the Department of Computer Science. Her main research interests are in software processes, software quality and software measurement. She has published widely in these areas.

Dr. Valerie Flynn is a Senior Lecturer in the School of Computing, Information Systems and Mathematics at South Bank University in London. Her research interests, and publications, are in the areas of IT education, decision-making and the management of technical expertise.