

Conducting Field Studies in Software Engineering: An Experience Report

Rebecca Yates

Lero - the Irish Software Engineering Research Centre
University of Limerick, Ireland
`rebecca.yates@lero.ie`

Abstract. Involving practitioners in software engineering research is crucial for relevant, applicable results, but finding willing participants can be difficult. This experience report describes some of the issues that can block or mar field studies of software engineering, and suggests tactics to avoid or mitigate these problems.

1 Introduction

Field studies of software developers at work are important for generating externally valid software engineering research results. Unfortunately, persuading software practitioners to participate in research projects can be difficult, and conducting the studies themselves has many potential pitfalls.

This paper reports on the experiences from a study of software engineering onboarding (i.e. their period of familiarisation with the company, the development team and the unfamiliar code and tools) (Yates 2011). This project collected recordings of onboarding sessions using a camcorder and screen capture software, interviews with the new developers, and questionnaires to capture the participants' background as developers. Out of around 70 approaches to a variety of companies, eight were willing to participate in some way.

The lessons learned from setting up and conducting the study are discussed here in the hope that this information will be of use to others designing similar field studies. Please note that the interpretations of events are only my opinion.

2 Finding participants

Having designed a field study, the first stage is to find willing participants. In this section, I discuss methods for reaching potential participants, understanding their concerns about participation, mitigating those concerns, and common responses from contacts.

2.1 Contacting developers

Personal contacts The most successful way to find participants was through personal contacts in industry, academia and other groups. The majority of the field studies were set up via friends and colleagues. As with job-hunting, spreading the word about the search can lead to useful introductions, which in turn can lead to field study opportunities.

University-friendly companies Some companies contact favoured universities as part of their recruitment drives. Companies who recruit in this way are more likely to be positive about academia in general and the institution in particular, so this is a good source of potential participants. As my research required software teams that were taking on a new member, I responded to recruitment emails sent to the university's computer science department. This strategy was occasionally successful.

Social networking Social networks (such as Twitter, LinkedIn and mailing lists) can be used to broadcast requests for participation. A colleague reported some success with this technique for a low-involvement request (completing an online survey), but it did not generate any responses to my more involved proposals.

Careers fairs Careers fairs provide a opportunity to meet face-to-face with company representatives. However, the representatives are typically either HR staff (who may pass on details of the research but are unlikely to be enthusiastic themselves) or junior developers (who may be enthusiastic but have little influence within their organisation). I had hoped that careers fairs would allow me to contact companies preparing to take on new hires, but this strategy did not prove a fruitful source of participants. If this approach is used, it is important to be clear that one is not actually looking for a position with the company, or the conversation can become very confusing.

Participant recommendations Initial participants can be asked to recommend others who might be willing to participate. I did not try this technique, but based on the comments I had after the data collection, I believe that it would be fruitful with participants who found the experience positive. In the case of this observational study, participants often commented that the involvement was much less onerous than expected, required very little of their time, and generated some valuable introspection; with these assurances, the recommended contacts might be more inclined to participate.

2.2 Common concerns

Software developers and their managers are likely to be enthusiastic about research projects that address problems they are living with, or opportunities to “show academia what the real world is like”. However, across all organisations, two major concerns prevent developers becoming involved with field studies: time pressures and security concerns. Addressing these issues upfront may turn an automatic ‘no’ into a constructive negotiation.

Time pressures While managers and developers may be broadly enthusiastic about a research project, often their major concern is that involvement will take up time (both during the actual data collection, and the overhead of arranging permissions, NDAs and so on). With many software development projects operating under tight deadlines, or already running late (see Perlow (1999)), any extra distractions may be considered unacceptable.

It is important, therefore, to be very clear about how much time the research will take up on top of the practitioners’ everyday work. Project managers will require time estimates for surveys and interviews, and a list of actions including administration (such as NDAs and visitor passes) which will also take time to complete. If the bulk of the data collection is fly-on-the-wall observation of developers’ normal activities, and therefore does not take time away from work, emphasising this may help to persuade contacts that participation presents little risk to their schedules.

Intellectual property Protecting intellectual property is sometimes a concern for potential participants (albeit not usually the primary concern). A non-disclosure agreement (NDA) is often enough to allow the research to proceed. Large corporations typically have dedicated legal teams and standard procedures for creating NDAs, while very small companies may not require NDAs at all (though ethically, the data and the participants must still be protected to the same degree). Medium-sized companies may struggle with this because participants see the need for IP protection but lack the resources or procedures to set this up. In all cases, NDAs may take longer than expected to set up, which may impact the research schedule.

IP is less of a concern when software is not the company's main line of business. It may be more comfortable for (for example) a materials supplier to allow access to its in-house stock management system, than for a software development company to permit published research on its flagship software product.

Project-specific issues The details of the project may introduce extra constraints on the search for participants. My research project focussed on onboarding, so I sought out software teams that were planning to introduce a new developer to their team in the near future. Many potential participants were enthusiastic about the project but unable to help because no new recruits were joining. This was sometimes even the case when the company was actively recruiting, because finding the right person could take many months.

2.3 Persuasion

A contact who is in a position to approve the research will have a set of questions about the requirements and implications of getting involved; these questions are often common to all organisations. However, if this person is not the initial point of contact, it can be difficult to communicate these details through a chain of command. Misunderstandings may lead to difficult situations where managers agree to participate, only to discover that they have actually been asked for something very different.

To mitigate this problem, the purpose and requirements of the research can be provided to the organisation on a single-page PDF (see appendix for a sample document used in my research). This provides a quick and accurate summary of the proposed research, which can be passed through an organisation from the initial contact to the correct person. The content of the information sheet evolved in response to frequently asked questions and occasional feedback.

This experience suggested that the most important points to be clear about were:

- The purpose of the research.
- The actions required from the participants and others in the company.
- The timescale of participation (in particular, any time taken away from work).
- The ways in which the collected data would be used and protected.

2.4 Responses

Common responses include:

That sounds interesting, but no. This is very common.

Sorry, we don't participate in research at Company X. This may simply mean that this contact is not interested in the overhead of organising participation. It may still be possible to conduct a study at this company via another, more enthusiastic contact.

An initially positive response, and then nothing. This may indicate that the company is too busy to participate, but occasionally a polite reminder will allow the conversation to move forward.

No response at all. This is the most common outcome. Occasionally, out of the blue they will reply months later (in one case to let me know that the NDA was set up and they were ready to help).

We would like to help but we are not in a position to do so. My study had the additional requirement that participating companies had new developers joining their teams. It is occasionally worth contacting them again after several months to see if the situation has changed.

Yes, we'd like to help, let's set up a meeting... It does happen.

Given the number of contacts (including multiple contacts for large companies) I found it useful to maintain records of contacts, particularly to highlight those that gave a positive response but had not arranged to participate. A spreadsheet function can show the number of days since the last communication, to ensure that timely reminders are sent and potential participants are not lost.

3 Technical issues on-site

Having negotiated access to a company, it is important to make the most of the opportunity and avoid wasting the participants' time. Most of the potential pitfalls can be avoided through good preparation.

Ambient noise Ambient noise can pose a problem for audiovisual recording. Open plan offices can be unexpectedly loud, and meeting rooms can have air-conditioning that nobody knows how to control. Even areas that seem quiet at first may be used at breaktimes for coffee, loud conversations and games of pool and pingpong. Depending on the nature of the field study, this may be unavoidable, but if the study does not require the participants' usual working environment, the participants may be able to book a quiet room for the study.

Some microphones are directional and can be aimed to capture more of the desired sound. Getting to know the audio equipment's capabilities beforehand will allow better recordings to be made. Depending on the purpose of the recordings, ambient noise can be frustrating, but is rarely a major problem.

Software installation and use As companies grow larger, they are more likely to hand control of their IT systems to dedicated teams. This can be an issue if the research requires the installation of any software (for example, screen capture or task recording). The IT department may have prevented the installation of any new software on participants' machines.

Occasionally, participants already have suitable software on their machines (for example, screen capture is currently available as part of QuickTime on Snow Leopard). If not, participants can be asked to preinstall the software, allowing them time to negotiate permission with their IT departments if required.

An alternative approach is to bring standalone ("portable") copies of the required software on a USB key (these do not require installation). In this case, the participants' OS may not be known in advance, so it is worth bringing different versions of the software to cover all eventualities.

It is very important to minimise the risk of negative effects of participation. Providing software is a risk because of the possibility of introducing viruses or other malware. Make very, very sure that any software you ask them to install or use is safe.

Running extra software on the participant's computer may cause it to slow down, impacting their interactions with it. In the case of screen recording software, the performance impact can be minimised by reducing the video quality and preventing unwanted audio recording.

Power sockets Power sockets can be a surprising issue. In many offices, every available socket is in use, or the only available socket is on the other side of a walkway. The activity to be recorded may take place away from any power sockets, and participants may move from place to place (e.g. to access whiteboards or demonstrate hardware). Again, familiarity with the recording equipment will prove valuable - for example, knowing how a camcorder will behave if unplugged during recording. It can also be a good strategy to bring an extension cable.

Data size Depending on the nature and length of data capture, data files may become very large. This creates several issues.

Data files can be kept small in the first place by choosing appropriate recording quality settings. Some screen capture software defaults to very high quality settings, and also records audio which may not be required.

During long sessions, the recording device may fill up before the participants have finished. Knowing the quality settings and the hardware specifications, it is possible to calculate in advance how much storage will be required (or how much of a session can be recorded at once). It may be possible to take advantage of coffee breaks to swap storage devices (e.g. spare memory cards) or transfer data files into storage, but this can be a lengthy process.

Data storage can be a particular problem if recording software is in use on the participant's computer and they do not have enough free space available. In this case, one option is to save the file directly to an external drive (this also avoids lengthy data transfer times after the recording session). An unexpected issue in this study was the 4GB file size limit on older Windows machines; files larger than this become corrupted and unusable.

Questionnaires As part of my study, I asked participants to complete an online questionnaire about their background as developers. Despite testing the questions with colleagues beforehand, some of the questions were difficult for participants to answer accurately. Examples include the question "how long have you worked at this company?". This apparently straightforward numeric question was difficult to answer for the developer who worked for the company as a contractor, then worked elsewhere, and then returned to the company as an employee. A free text box often allows more intelligent answers.

In another scenario, the questionnaire had to be hastily adapted because a non-developer became involved in the recordings and the questions were worded with the assumption that all participants would be developers.

4 Ethical issues of collaboration

Sharing data It is important to be clear with the participants and the company about who the recordings or other data will be shared with. Typically, recordings should not go beyond the participants and the research group, but occasionally other employees in the same company may ask to see or hear them. This could be an issue if the participants, believing the data collection to be private, have been discussing other employees or office politics, or making jokes that could be misinterpreted. Sharing the recordings could have negative consequences for the participants; this must be avoided.

Participant consent Information does not always flow as freely through corporations as one might hope. It is entirely possible to set up a field study with a manager, and arrive to discover that the participants themselves have not been informed or consulted about the planned research. This presents the ethical issue of informed consent; participants may then feel pressured into participating.

This scenario could be avoided by requiring potential participants to complete an online survey or other task before the recording session. This would ensure that they were aware of the field study and able to discuss any concerns beforehand.

Unexpected participants During a field study, it is fairly common for extra employees to interact with those who have agreed to participate in the research. A pre-prepared explanation and spare participant consent sheets provides the option for them to join the study, if they wish, without major disruption to the session.

5 Thanking participants

It is often recommended to “give something back” to the companies involved in the form of presentations about the research. So far, not a single company has taken me up on this (probably because of the time pressures mentioned earlier).

As thank-you gifts, physical Amazon gift vouchers were appreciated, but electronically sent vouchers were often mistaken for spam. A thank-you card and a box of chocolates or biscuits were appreciated in office environments.

6 Related discussion

Further discussions on ethical issues can be found in the special issue of *Empirical Software Engineering* focusing on ethics, edited by Singer & Vinson (2000), and data collection techniques for field studies are examined by Lethbridge, Sim & Singer (2005).

Experimentation in the field does not always go to plan. Exton, Avram, Buckley & LeGear (2007) provide a discussion of technical and social issues that affected an experimental setup.

7 Acknowledgements

This work was supported, in part, by Science Foundation Ireland grant 04/CE2/I303_1 to Lero - the Irish Software Engineering Research Centre (www.lero.ie). The author thanks Jim Buckley and Norah Power for their comments and advice.

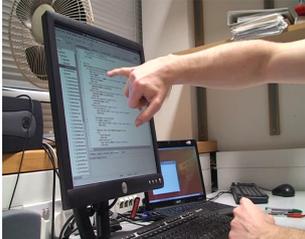
References

- Exton, C., Avram, G., Buckley, J. & LeGear, A. (2007), An experiential report on the limitations of experimentation as a means of empirically investigating software practitioners, *in* ‘Psychology of Programming Interest Group’, pp. 173–184.
- Lethbridge, T., Sim, S. & Singer, J. (2005), ‘Studying software engineers: Data collection techniques for software field studies’, *Empirical Software Engineering* **10**(3), 311–341.
- Perlow, L. (1999), ‘The time famine: Toward a sociology of work time.’, *Administrative Science Quarterly* **44**(1), 57–59.
- Singer, J. & Vinson, N. (2000), ‘Ethics and empirical studies of software engineering’, *Empirical Software Engineering* **5**, 89–91. 10.1023/A:1009859121816.
- Yates, R. (2011), Expert explanations of software, *in* ‘Psychology of Programming Work in Progress’.



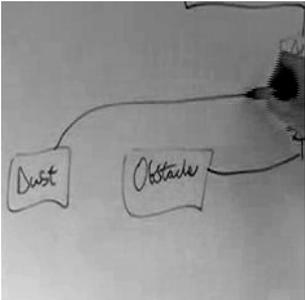
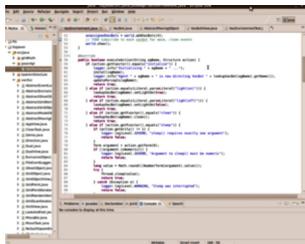
The Issue

- On joining a company, or moving to a new project, a software developer is faced with unfamiliar code and has difficulty contributing. Ramping up to full productivity can take up to six months.
- In research at Microsoft, 56% of developers reported that understanding someone else's code was a serious problem.
- Expert software developers are able to mentor new team members – but the experts aren't always available, and mentoring can be time-consuming.



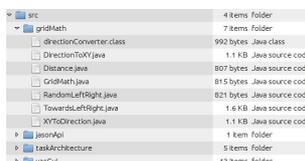
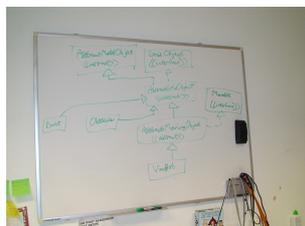
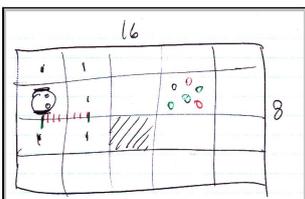
Research Proposal

- To address this issue, I'm investigating how experts explain their code to new team members. Through video recordings, screen capture, code analysis and interviews, I'm seeking to understand what information about a codebase is most useful in this situation.
- This is a new approach to the problem. To ensure valid and useful results, it's important to involve industrial partners.
- More information about the data collection can be found on the reverse.



Outcomes

- This kind of research can answer questions such as:
 - Which parts of the code are important for a new team member to know?
 - How are diagrams used to convey information?
 - How useful was the explanation to the newcomer?
 - How could the ramp-up be improved?
- I hope that this research will lead to recommendations for mentoring techniques or tool support for new team members, reducing ramp-up time and increasing developer productivity.



If you have any further questions, or would like to participate, please contact me: (rebecca.yates@lero.ie). Thank you for your time.

Rebecca Yates is a PhD student at the University of Limerick, Ireland, supervised by Jim Buckley and Norah Power. Prior to this, she worked as a software engineer.



Thank you for considering helping with this research into expert explanations of software. This page details the steps typically required for participation.

Beforehand:

We arrange a time for the data collection session.

I am most interested in a newcomer's first introduction to the codebase. The walkthrough typically takes between one hour and half a day. Newcomers may be new hires, or existing employees moving to a different project within the company.

We set up an NDA if required.

This can be supplied by your company or created by the University of Limerick.

On the day:

The participants read and sign the Participants Rights document.

This document covers their rights regarding anonymity, their right to pause or abandon the recording for any reason, and describes how the collected materials will be used.

We install screencapture software on their machine.

e.g. the freely available CamStudio (unless a suitable program is already installed). This is because the text on screen doesn't show up clearly on the camcorder.

I place the camcorder and start recording.

This is a small camcorder on a desktop tripod, and is pointed at the screen. It's unobtrusive, and can be moved easily to follow the conversation to whiteboards or other locations.

The participants work as usual, with the expert explaining the code to the newcomer.

I move the camcorder if required, and don't interfere with the work in progress.

After the session, I take a photo of any diagrams that were used.

Like the screencapture, this just ensures a clear copy in case it's not readable on the video.

I analyse the source code under discussion.

This is for the quantitative part of the research, comparing the path of the explanation to the code metrics. All copied materials are stored encrypted using ecryptfs and are not released; if code cannot be removed, on-site access would suffice.

Follow up:

The participants complete an online questionnaire.

This is a short (~10 minutes) questionnaire on their background as software developers.

The newcomer gives a short interview.

This takes place around three weeks later, and asks how useful the explanation was. This can be completed in person or by phone and should take less than half an hour.

Optionally, I can provide a report or presentation on this research.

To respect the privacy of the participants, this is a report on the research in general, and does not name any individuals or companies. As detailed on the previous page, this report would focus on parts of the code base that are important to new team members, how diagrams can be used effectively, the usefulness of different types of information and general improvements to the ramp up process.

