The role of ethnographic studies in empirical software engineering

Prof Helen Sharp,
Centre for Research in Computing
The Open University, UK
h.c.sharp@open.ac.uk

EASE 2012
About The Open University

Computing Building, Milton Keynes, UK

In addition, 13 regional centres are spread around England, Scotland, Wales and Northern Ireland.

• The OU is the largest academic institution in the UK and a world leader in flexible distance learning. Since it began in 1969, the OU has taught more than 1.5 million students.

• More than 21,000 OU students study outside the UK.

• The OU has around 260,000 students studying at any one time.

• Software Engineering research ranked fifth in Europe.
Overview

- Context and one example
- What is Ethnography?
- Three ethnographic studies of software development
- Strengths and weaknesses
- Some pitfalls and rewards
- Common myths and expectations
- The role of ethnographic studies in empirical software engineering
Overview

• Context and one example
• What is Ethnography?
• Three ethnographic studies of software development
• Strengths and weaknesses
• Some pitfalls and rewards
• Common myths and expectations
• The role of ethnographic studies in empirical software engineering
People and software

The need to account for developers and their interactions is not new, for example:

ICSE 2012 has two tracks on human aspects and one on social aspects and a workshop in its 5th year
Programmer personas

• Observations of practitioners (developers, testers etc)
• Photos, artefacts, paired observations
• After multiple observations, patterns emerge
• Mort, Elvis and Einstein have their own Wikipedia page
• 19 personas were in use a year ago
• Observations of testers led to ‘Ellen’ (2006)
  “we needed to create a completely different experience to the standard Visual Studio experience”
Why ethnography is useful for empirical software engineering

• Because software practice involves people and their interactions

• Because of what we do
  – Empirical software engineering aims to improve software practice through tools, methods, processes... AND
  – An essential prerequisite to improving practice is to understand it
Why ethnography is useful for empirical software engineering

• Because of the realities of software practice
  – What interviewees say and what practitioners do
  – What is documented and what happens in practice

• Because it may be more persuasive to practitioners?
  – Practitioners are influenced more by peers than by research
  – Rich data set grounded in practice may be more convincing
Overview

- Context and one example
- What is Ethnography?
- Three ethnographic studies of software development
- Strengths and weaknesses
- Some pitfalls and rewards
- Common myths and expectations
- The role of ethnographic studies in empirical software engineering
What is Ethnography?

- Ethnography is a qualitative approach to research which aims to observe activity, to understand the informants’ point of view, and to make the implicit explicit.
  - The main focus is on the informant’s point of view: what is and is not important, relevant, interesting, painful, exciting **to the informant, not to the researcher**.
  - In traditional ethnography, the researcher aims to gain this understanding and write about it; writing is as important as anything else.

- Greek word
  - *ethnos* = nation
  - *graphein* = write

→ Writing a culture
Ethnography’s origins

1915: Bronislaw Malinowski’s “Argonauts of the Western Pacific”

- Field studies should be in the field, not in a library as was done before
- Focus on exotic, “primitive” cultures, on understanding institutions, costumes and daily life
Ethnography adopted in CSCW/HCI

- Nardi (1990): spreadsheet users
- John Hughes, Bentley, Randall, Rodden, and others (1992): air-traffic controllers
- Julian Orr (1996): copy-machine technicians
- User Experience designers routinely use ethnographic techniques now
Key principles

• Aim to identify informants’ point of view
• Acknowledge and account for your own assumptions and background
• No *a priori* expectations
• Non judgmental
• Non intrusive
• Data analysis is systematic and on-going
• Seek confirming and disconfirming evidence
Main techniques

• Make implicit explicit, so treat the familiar as strange
  – Be surprised
  – Why are activities done that way?
  – Who is involved in activities and who is not?
  – Stop and reflect

• Ethnographic ‘interview’

• Informant checking

• Participant observation
  – Spend time with your informants
  – Observation plans change continually
Participant observation

“Ethnography's hallmark is this notion of participant observation, the idea that you learn about other people's cultural practices by going there, being there, and by doing it with them.”

“... as human beings <we> try and make sense of data points from very limited perspectives. You give us two data points, and we want to immediately jump to the story that makes those two points make sense... to connect those two points ... using a story line ... you're already familiar with ... Participant observation allows you ... to get to the point where you know the stories they would use”
What data to collect?

The following is an illustrative list:

• Documentation: activity or job descriptions, rules and procedures said to govern particular activities.
• Descriptions of activities observed.
• Recordings of the talk taking place between parties involved in observed activities.
• Informal interviews with participants explaining the detail of observed activities.
• Diagrams of the physical layout, including the position of artefacts.
• Photographs of artifacts used.

Sometimes developers wrote out a new card that was pink in colour and placed it on a separate piece of wall, labelled ‘Bugs’ with other pink cards. People tended to notice this or remark on it.

Example data
Suitable research questions

‘How do software practitioners develop systems using XP?’ rather than

‘Is single programmer coding more productive than pair programming?’

‘Why don’t scientists adhere to a company manual of software development practice?’ rather than

‘Does structuring the manual this way help scientists produce more lines of code an hour?’

‘What are the characteristics of a technology adoption?’ rather than

‘How did the ideas of Simula develop into Java?’

Adaptations of traditional ethnography

• Virtual Ethnography (online)
• Netnography (marketing)
• Ethnography in distributed settings
  – Trace ethnography (following artefacts)
  – Multi-site ethnography (choosing a vantage point)
• Contextual inquiry (apprenticeship)
Overview

• Context and one example
• What is Ethnography?
• Three ethnographic studies of software development
• Strengths and weaknesses
• Some pitfalls and rewards
• Common myths and expectations
• The role of ethnographic studies in empirical software engineering
Three (more) ethnographic studies

1. **Freshwater**: more traditional ethnographic study, one company, written ethnography as output

2. **Scientist end-user developers**: longitudinal study of one project (involving several organisations) that moves beyond ethnographic studies to use other research methods

3. **Agile software development**: group of studies in different companies, but within the same community (i.e. agile community)
Freshwater

• Research team:
  – a sociologist, a computer scientist and a management scientist
  – took a ‘sociology of scientific knowledge’ view
• Ethnography and discourse analysis
• Participant observation
  – management scientist spent 15 months on the development team
  – 4 days a week working as a software engineer

Janet Low, Jim Johnson, Pat Hall, Fiona Hovenden, Janet Rachel, Hugh Robinson, Steve Woolgar (1996) ‘Read this and change the way you feel about software engineering’, Information and Software Technology 38 77-87
Freshwater ‘Findings’

• Multiple stakeholders are present in a software project, each with a different perspective. These different perspectives recognise ‘progress’ differently.

• The influence of the life-cycle and the methodology were pervasive. While they may not constitute a single narrative, they do offer one which is both pre-eminent and deeply entrenched.
Scientist end-user developers

- Project manager approached researcher because the project was having problems
- Researcher performed observations followed up by other research methods, but maintained ethnographic focus
- Participant observation
  - Several key meetings
  - Close contact with software project manager
- Overall involvement already over 4 years

Scientist end-user developers

- Expectations confounded
  - Project manager thought the problem was due to the distributed nature of the project
  - Researcher suspected it was due to clash of software developers and scientists
- Articulating the different values, assumptions, and habits of the two groups had a very positive effect
- Making them explicit helped to save the project
**Agile software development**

- Ethnographic approach
- ‘Participant’ observation of > 10 teams
  - Observation length of at least one sprint/iteration
  - Companies from different sectors (banks, web developers, software tools, document preparation)
- Seeking to understand how and why agile ‘works’
- Focus on people and interactions

Agile ‘Findings’

- Generated questions for further investigation, e.g. physicality
- Provided context for code analysis
- Implicit exposed, e.g. size of cards, responsibility of individuals
- Potential challenges for UX and agile combination

Overview

• Context and one example
• What is Ethnography?
• Three ethnographic studies of software development
• Strengths and weaknesses
• Some pitfalls and rewards
• Common myths and expectations
• The role of ethnographic studies in empirical software engineering
Strengths and weaknesses

Strengths:
• Implications for academia and practice
• The Power of ‘being there’
  – Experience speaks louder than report

Main weakness:
• Ethnography can be difficult to apply
  – Researching ‘up’
  – Need to deploy the understanding gained towards improvement
  – Initially researcher can feel overwhelmed
Some pitfalls and rewards

Pitfalls:
• Having no focus
• Not being flexible
• Being judgmental

Rewards:
• Rich data based on real activity
Common myths and expectations

• Rigour
• Quantification myth
• Interviews give the same information
• Generalisation
• Relationship with informants
• <Ethnographic studies last months or years>
“Ethnographers are noted for their ability to keep an open mind about the group or culture they are studying. This quality, however, does not imply any lack of rigor. The ethnographer enters the field with an open mind not an empty head.”


“The objectivity of empirical research comes from a review process that assures that the analysis relies on all the relevant evidence and takes into account all the rival interpretations. This can be done (or not done) in both a quantitative and a qualitative analysis”

Quantification myth

• Quantitative data is ‘more rigorous’, ‘right’, ‘stronger’
  – precise but not necessarily accurate
  – interpretation is limited (Capiluppi et al 2007)
  – statistics (non-parametric)
  – generalisation to a population (what population?)
• Ethnographies can only result in qualitative data

Why not just use interviews?

• People have a habit of:
  – telling you what you want to hear
  – creating a rationalised account which may not reflect everything you want to know about
  – using selective memory (for their own purposes)

• Not malicious, just natural
Generalisation

“[Even if] one swallow does not make a summer, ... one swallow does prove the existence of swallows. And careful dissection of even one swallow may provide a great deal of reliable information about swallow anatomy"

Relationship with informants

• As a software engineer, you are already part of the same community
  – Easier to talk to and easier to understand
  – Harder to view the familiar as strange

• Observer effect
  – “today’s not a typical day”

• Gaining access can be hard but
  – Identify what the informants will gain
  – Be professional and clear in your relationship (ethical issue)
  – Informed consent? (ethical issue)
Overview

• Context and one example
• What is Ethnography?
• Three ethnographic studies of software development
• Strengths and weaknesses
• Some pitfalls and rewards
• Common myths and expectations
• The role of ethnographic studies in empirical software engineering
The role of ethnographic studies in ESE

- To inform tool and process development
- To unearth more specific research questions
- To develop a deeper understanding on which to build improvements – for theory and for practice
- To complement other approaches (e.g. code analysis, quantitative studies) by providing context & understanding grounded in real practice
- To strengthen investigations into the social and human aspects of software engineering
MINING FOR DEEPER INSIGHTS

I THINK WE BROUGHT THE WRONG TOOLS.

SO WHAT?
BEEN THERE... SEEN THAT.
LIES... ALL LIES!
GOLD!

Any questions?