Stories from the Mobile Workplace: An Emerging Narrative Ethnography

J. Hunt, P. Romero, and J. Good.

IDEAS Lab, Informatics Department, University of Sussex
Falmer, Brighton, BN1 9RH, UK
{j.m.hunt, pablor, j.good}@sussex.ac.uk

Abstract. This paper presents preliminary results from a narrative-ethnographic field study of a small mobile software development company. Initial ethnographic findings are presented, and further narrative analysis is proposed in the areas of stories relating to programmer stereotyping, war stories, and humour. It is expected that the study will form the first case-study for a broader comparative investigation into how stories are used and told across software development methodologies.

1 Introduction: Communication Issues in Software Development

This paper presents preliminary discussion, and plans for future study, arising from a narrative-ethnographic field study of a small mobile software development company. It presents the initial ethnographic data, focussing on observed communication. Further analysis planned from the perspective of narrative research is then outlined.

The work discussed in this paper reflects the context of increasing recognition of the importance of research into the nature of programmer dialogue and interaction within software development practice. This is particularly demonstrated by increased interest and research into communication and interaction in Agile Development Methodologies and associated techniques [5][35].

Software development is a team process. To produce non-negligible software programmers must collaborate and communicate. Arguably one of the major problems faced by any software development endeavour is sociological in nature, as distinct from technological [8]. In one sense "...the programmer working alone is working under a serious handicap" ([42] p. 46). For example, in the case of Agile Systems Development [45][6] there is a recognition that the "most efficient and effective method of conveying information to and within a development team is face-to-face conversation" although working software continues to be the primary measure of progress. This view of programming as a social process necessitating communication is a value statement agreed by all agile software methodologies and can be contrasted with traditional views of software development as a formal engineering process. This shift has been described as the breakdown of the modernist narrative of software engineering [33].

2 Motivation and Method: Narrative Research and Ethnography

Narrative Analysis [30][15] is a qualitative research technique which focuses on the ubiquitous ways in which people make and use stories to interpret the world. This is achieved through the analysis of narratives as chronologically-told discourse elements. Narratives are considered to be social products within specific contexts, and an interpretive device through which people communicate knowledge and define their own identity. It is arguable that the technique is particularly useful for sensitising a researcher to ethical and cultural ambiguities. Any narrative investigation is fundamentally aware that "stories are not produced in a vacuum, but their telling is always situated within an interactional and sequential context" ([14] p. 131) and they "are situated, and it is this very situatedness within social life that militates against conceiving of them in transcendent terms" ([9] p. 106).

The approach is starting to find popularity outside of social science [30][16] and organizational studies [7][4], and is beginning to be used by the information systems community, especially for investigating tacit knowledge transfer and learning [17]. A strong case for narrative research in collaborative systems research is made by Pentland [27]. Tan and Hunter have used a narrative approach to look at career progression in information systems professionals [39]. Alvarez and Urla [1] have applied narrative research in the context of knowledge elicitation interviews. Narrative semiotic techniques have been used to investigate use case analysis [19]. An overview of some different approaches to narrative research in information systems development is broadly given by Wagner [41].

There is increasing recognition that the most basic and prevalent form of narrative actually arises as the product of ordinary conversation [23][11][25]. This type of story is labeled as a "small story" [12]. One example of what constitutes a small story is the case of troubleshooting stories, such as those studied by Julian Orr [26] as part of his ethnographic study into photocopier repair technicians. He states that such stories become "artefacts to circulate and preserve. Through them, experience becomes reproducible and reusable" and they are "told in pursuit of more purely social functions than diagnosis. They preserve and circulate hard-won information and are used to make claims of seniority within the community. They also amuse, instruct and celebrate the tellers' identity as technicians" ([26] p. 126).

This type of narrative research combines well with observational and ethnographic methods of data collection, especially when considering "small stories." Ethnographic research [44][40] encourages researchers to immerse themselves into the culture which they are studying over a period of time. This style of research has proved an increasingly popular approach for considering software development in information systems [38] as well as computer-mediated communication [22]. Most notably an ethnographic approach has recently been taken by Robinson and Sharp [32] for exploring culture in mature XP teams.

The two approaches detailed in this section were combined in order to investigate a small software development company. The approach, although ethnographically similar to the work of Robinson and Sharp, does not attempt to take an objective approach to the data and analysis is focussed on small story and narrative elements. Presented in the next two sections is the preliminary ethnographic data and analysis.

3 Narrative Ethnographic Study

The discussion in this paper draws on research from an exploratory field study which has been recently been conducted with a small mobile software development company, specialising in design and technology for the mobile internet. The research incorporates narrative interviews supported by observational data. The original intention of the study was to observe what and whether stories were told by developers, and what that means in relation to their working environment and the production of software. Although the selection of the company was opportunistic it directly leads into further study in the area. This is discussed further in Section 6.2.

3.1 Field Study

The initial field study occurred over a two-week period and was carried out in December 2005. The company had just celebrated its fifth birthday. The aim of the initial field study was to observe staff meetings and the mobile software company at work. Semi-structured narrative-elicitation interviews were conducted as a follow-up investigation with all staff members three months later. Extensive field notes, recordings, collections of documents and photographs, and a diary were generated. It should be noted that the research is not strictly ethnographic in a traditional sense, as it does not involve full-immersion in the culture over time (generally for months to years). However, it does directly use techniques and approaches typical of the technique.

The company was based in a modern, shared open-plan loft space area which was shared with two other small companies. The group appeared calm and capable when observed at work, with occasional banter, conversation and shared humour. The company undertook interaction design, prototyping work and full product development for clients, and specialised in mobile applications. All software was developed in-house and the only outsourced work was application testing (although this was managed in-house by hourly contractors). The company used a relaxed form of Scrum [34] for development practice. Most of the employees were still in the early stages of their working career, but the group had a strong bond between members and a clearly stated belief in Human-Computer Interaction values and designing for usability.

The period observed was one where one member of staff left and another joined. The company was observed at work, and five meetings in total were directly studied; two weekly management meetings, two weekly group meetings involving all members of permanent staff and a project meeting.At the start of the study there were nine members of staff (including the company director). This reduced to eight by the time of the follow-up interviews.

3.2 Narrative Interviews and Professional Career

Three months after the initial field study semi-structured narrative-elicitation interviews were conducted. These were carried out over a period of twelve days. All staff members, at this time eight in total, were interviewed. Their interviews lasted between 40 minutes and 1 hour 20 minutes. Only one question was directly asked about events important to them in terms of ‘professional career’. Further information, elaboration of points and specific narratives relating to interest points from the observation phase, were then elicited by the interviewer. Subsequent questions asked only used words taken directly from the interviewees’ responses, and generally in the same chronological order, in order to elicit further stories about their career in software development and to investigate further themes.

It was expected that these interviews would naturally support observations made during the initial field study. This was because the interviewees were free to talk about whatever was a concern to them, or interesting to them throughout their professional career, and therefore any repetition of themes previously observed in the field study occurred naturally. Responses ranged from detailed discussions of their current employment experiences through to progressions from ‘first computer’, or first part-time job, leading to their current employment within the mobile software industry. In many cases, the move to mobile computing was detailed as a positive move as part of the working life related journey; away from an unstable background in web-development, as an early job out of academia, or away from the strains of commuting. Interpersonal concerns, working issues (including uncertain, still-merging working practices), as well as personal employment histories, were also narrated.

4 Initial Findings and Storied Themes

This section broadly presents some initial observations of themes from the ethnographic data, as well as preliminary narrative analysis from both the field study and the narrative interviews. The majority of data is taken from the observations of the company and the group meetings, as well as the narrative interview of the new staff member.

4.1 Shared Space and “Magical Tables”

The office space was located at the entrance of the top floor of a large converted warehouse. The space was light, airy and very open plan. It was shared with two graphical design companies. There was a small kitchen area near the entrance and a group meeting area at the other end, which was shared by all.

Due to company growth, the majority of team members now worked on one long desk. An island desk was nearby on which the company director and the client services director were seated (See Figure 1). Group communication was mostly face to face - the layout of the office was conducive to this, although

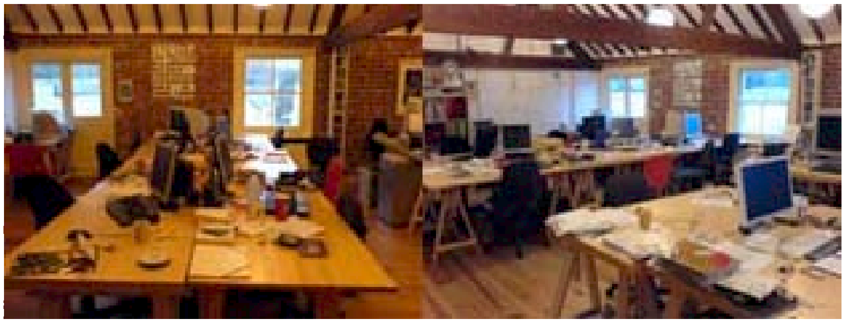

Fig. 1. ‘Last Supper’ Desk and Management desk view over to ‘Last Supper’ Desk

direct communication between the company director from his desk to other staff was observed to often involve necessarily talking over staff members. This can be seen from the photos of the office environment included in this paper.

The long desk layout with the director on the outside was recognised as not ideal. Issues around mess were described in terms of ‘programmers’ being messy and ‘designers’ liking things to look nice, and in relation to the lack of storage space. As was requested in one meeting: “Can you try and keep your desks tidier pleeease? They do get rather messy. Jungle-like.”

Stories were told in interviews about the ‘magical desks’ which were planned to save the company from the problems currently being caused by the ‘last supper’ table, and which were being planned to optimise group communication. As one person states “That’s quite interesting . . . in the fact that everybody thinks these new tables will magically resolve a number of issues. They will be the magical tables. I’m not so sure.” He continues “I think it would help when you are working on a project.” “I don’t think it will help with ‘messy’ issues” but “it will make things nicer I think.”

Team interaction was specifically noted in the second week, when it became clear that the new team member was only slowly adjusting to the open plan office. The relaxed style of meeting and open space appeared to make boundaries hard to distinguish. On his third day on the job he walked up to the management meeting with a pad and pen and made to join them. He then had to be informed of his mistake. The open plan design also meant that when brief meetings were conducted outside of the designated meeting space it was often hard to tell when such meetings were being formally conducted. This also wasn’t helped by the informal, joking, working environment.

Issues around noise were mainly due to the fact that one of the other companies sharing the space worked with a radio on at all times. This meant that everyone was listening to Radio 2 at all times. It entered conversations regularly; both in meetings and during the average work day where people would start singing along or discussing a particular singer. This was anecdotally referred to as a problem which was caused by a difference between programmers and designers; It was indicated that ‘designers like noise’ and ‘programmers like quiet.' In this instance 'designers' referred to graphical designers and such statements were never discussed in relation to team members who had responsibilities for design aspects as well as pure programming.

4.2 Group Membership

The team was observed on initial visits to be comfortable and relaxed while working with each other. This is informed by the observations of shared humour, bonding through food, and shared technology which are presented in this section. Meeting styles were very relaxed, and humour played a central role.

Group Technology The company used two main collaborative tools; a web based debugging tool to keep track of operations, projects and tasks, and a company wiki to keep important detailed summaries and for staff members to establish their own personal space within the company. The former was extensively used in meetings and the latter was only occasionally referred to. This was partly because it was still under development, but there were hopes it would continue to grow to become an important resource for the group so they could use it to integrate people into the company, as a personal space, as an area for standard policies and documents, for details of personal projects and for marking the closure of projects in terms of lessons learned.

The company was trying to establish techniques for introducing new team members to working processes etc. It was hoped that this will be achieved through further developing the wiki as a primary induction tool, although it was not yet directly used for this purpose due to the small size of the team. Induction processes were still unfixed and unclear.

"We Love Food" The researcher was told by several members of the team that it was strongly bonded by its love of food, and that that was a common topic of conversation. This was encouraged both through the use of a dedicated 'food' section of the wiki and through a weekly group lunch. The dedicated 'Food Page' on the company wiki states:

> Breakfast
> 1. l33t breakfast casserole
>
> Lunch
> 1. Cheese
> 2. Cheesy Eggy Hammy
>
> [. . .]
>
> Dinner
> 1. Cheese
> 2. Cheese
> 3. Cheese

Few references to food were actually observed in interaction, although discussions were made in a team meeting of the unknown spirit which had been brought back from the Czech Republic for the group by a staff member and a set of interesting discussions were also observed while the company was at work, around using dishwashers to cook food. It was apparent that agreement on such matters was not always perfect:

> CD: So should we try to grab something to eat first?
> DESD1: It's bad enough trying to find a place to go for lunch.

Group Humour and Stories for Design The group were also bonded through shared conversational jokes. A prime illustration is this conversation which occurred during a full-group meeting. In this example the group are discussing bad advert design in relation to a website design previously produced by another company:

> DEV2: Is it going to be as bad as the advert on . . . ?
> CD: Hopefully worse. I think they seem to be sticking with the cock theme.
> DEV2: Yeah. [Group Laughter]
> CD: [to researcher] Oh, We'll have to show you.
> DEV4: It's not there anymore, it's gone. I looked, I tried to show my friend.
> CD: I think they're moving it onto the front page of the site now. They liked it so much.
> DEV2: One of my personal design philosophies is to always have a cock on the front page of a site.
> CD: Yeah. Whenever I'm looking at design I'm thinking 'but where's the cock?'

Staff members were observed as seeming free to voice personal opinion and this relaxed and natural meeting style was also exemplified in later discussions about naming a project product, where one person was accused of just trying to sneak in obscenities under the guise of naming a mobile technology application. This relaxed style appeared to actively encourage open discussion and can be considered indicative of comfort and a lack of perceived stress within the group.

One project was specifically picked up for discussion in relation to developing iconic interfaces for mobiles. A story was told about a sign in Prague:

> DEV2: I saw a really great sign in Prague. Large blue square with a white outline. So. Y'know. Information that it had. It had a house. And a car. Adult and a child and a football. And it told me absolutely nothing about anything at all.

This led to further discussion about what use of icons is made in airports, why they are effective whereas the example is not, and how it can be considered in relation to design for usability of mobile applications.

4.3 Narrating a Stereotype

This section briefly outlines how, for the new staff member, a culturally dominant narrative can, and is, used to counter and explain problems which are faced within the workplace.

During the observation of the first group meeting, during the initial field study, the company director announced that a new staff member would be joining the group in four days time, and that he would have an awful lot to learn in a very short space of time: He states "Two nightmare-ish days."

From the new employees' narrative interview, conducted three months after he joined, these first two days were described as leaving him "having to develop a project" using technology and techniques with which he was unfamiliar and adjusting to the "large changes." He says: "I didn't know what I was getting myself into with the job and I started erm gave my notice in and started the day after I finished." He further states "I was confident that I could do it and I was kind of wrong."

In relation to the two day handover he adds: "two days is ridiculous, you can't, you can't absorb that kind of information from someone, the kind of information that is not written down. Erm you might not know, but it's assumed. There's no way you should be able to do that."

In this situation the narrator was introducing himself into a group which was already closely socially knit, and into an environment which to him was "very stressful, very full on, lots of things to do that you've only got a short time to do so. Th-they want not only do they want quality but they want quantity and they want it done in a very erm short space of time."

He refers to his technical ability; he describes himself as the "best programmer" despite recognising some of his co-workers may be "a lot brighter than I am" and "are all very socially inte-intelligent." However this is stressful as he finds himself in a position where he says he wants to "beat them." This works against the reality discussed in the introduction where collaboration generally works better when one team member is not trying to "beat" the other members of the team. The lack of sensitivity to unspoken messages combined with the concepts of intellectual or technical superiority can be potentially detrimental to any team project [21].

Detailed analysis supports an interpretation that he is transmitting information consistent with the computer programmer stereotype [18]. This stereotype can be summarised as a person who is interested in technology, has strong technical skills, and who, in a stronger sense, may also be low in social skills. It can also be interpreted as a derogatory term.

He talks about how he "wrote a giant", a proof of his intelligence and technical capability as a problem solver under pressure: "I've got to make this deadline so I wrote this amazing task scheduler to the system." It was "amazing," "very, very, very clever" and a "giant." However, it initially failed as a project as he "thought [the client] understood the concepts" but he found out that "they didn't." Here the failure of initial implementation is framed in terms of failed communication and lack of understanding on the side of those he was collaborating with: "I thought they understood from all the conversations I'd had." The use of this phrase is particularly telling where the sense of one-way communication is highlighted by the speaker.

In the personal history given in the interview, speech and communication is a recurring problem. It gets "people's backs up" and leads to people losing their jobs and the company going "down the hole." "Career manipulation" was thus defined as not-communicating, a belief which has led to problems in the new place of employment where his co-workers are all "socially intelligent" and a place where he has been told "when I learn communication I'll be worth my weight in gold."

Such a description of self as a stereotype can be seen as serving to explain a failure in team cooperation. It may be interpreted in relation to Goffman's investigations of protection of self and the use of strategies which "give the mark an out; in case of failure he can act as if the self that has failed is not one that is important to him" [13]. The stereotype can be a good programmer even without communication skills, and can be part of the set of people who make "giant" and "grand schemes" without being a member of the current team. It is against this failure to bond, this realisation that he might have pushed his quietness a "bit too far" in his current employment that his entire story could be viewed to be situated. He is defended against never becoming "worth [his] weight in gold" and is appeased by the knowledge that the failure in terms of this project "wasn't because of [the] code [which was] kind of good."

It may be just a fictive construction that "bears no real reference to actual lives" ([29] p.314), but the interpretation of the text as given here is part of an ethnographic narrative attempt to, according to Geertz, "'figure out what the devil they think they are up to.' ([10] p.58)" [24].

4.4 Changing and Growing

During the second week of observation, the group meeting introduced several new formalisations and structures to the way people work; including holiday booking, admin standardisation ("We are sort of reinventing the wheel a lot on these things both visually and in terms of functions" "So everything we do can be done quicker and more consistently"), out-of-hours support structure, personal projects and greater clarity in the way projects are started up and completed.

The lack of company documentation was a concern of more than one employee, especially in terms of lack of end of project documentation. There was also concern to ensure that the end of projects should be properly marked and celebrated for company morale. These issues are also strongly reflected in the narrative interviews. As one person states "I just think we can just make things much more efficient." He goes on to say that documentation is not always easy to maintain in a wiki as "It's all too easy to put things in unstructured."

The introduction of the concept of a support team to deal with server issues etc. out of the office was interestingly negotiated. The discussions about having company calls diverted to personal handsets led to a story about how best to professionally answer the support phone - a concern when acting in an official role outside of the office. This again was discussed in terms of conversational humour within the company:

> DESD1: Erm. I'll have to get out of the habit of picking up my phone and ... like this morning I thought it was my brother so I answered it in a very particular way. Which wasn't particularly good.
> CD: In a 'particular way'?
> DESD1: Yes.
> DEV2: Which particular way would that be?
> DESD1: Morning Dick. [Embarrassed laughter]
> CD: Who was it?

DESD1: It was _'s boss.
CD: We need to work through that.

The introduction of personal 'gold card' projects (also seen in the ethnographic study in [31]) were also made in the same meeting. This would allow team members to work on their own projects for a portion of the working week if they were of potentially also value to the company. This had clearly been previously discussed by the group but not yet conveyed to the new staff member - so the idea was introduced in relation to the same concept being used at Google. Proposals for personal projects were introduced and discussed. People closely tied their personal projects to the interests of the company - apart from the new staff member who was instead interested in mimicking code. On later observation he had changed his chosen project and it was closer to the interests of the group.

A particular concern for several members of the company appeared to be a fear of growth. Mention was made that there were hopes that the company would settle at this size for a while longer ("Hopefully we'll just stabilise"). There was a strong sense that people generally enjoyed their work within the company and were proud of their work. This was also supported by the interview data, where the career move to the company was consistently presented as a positive one.

5 Continued Analysis

Further analysis of data needs to be conducted on the narrative aspect of this study, emerging and building from the ethnographic data collected as presented above. The three main types of narrative in the data to be considered will be briefly discussed next.

5.1 Stereotyping for Programmer Interaction

As discussed in Section 4.3, one feature observed in initial analysis was that a new team member was finding it difficult to adjust to the working environment. The narration works in opposition to the shift in view of agile systems developers as capable communicators. A programmer who lacks social skills "appears to be much less competent in the eyes of agile methods problem solvers, however. The problem solver isn't as sensitive to the unspoken messages other group members send" ([37] p.119). In this way, programmers with low social skills may potentially be marginalised by agile practice. This aspect shall be developed further, as it provides an interesting perspective on the agile approach, where communication skills are just as vital as technical ability.

Emerging from the ethnographic data there were repeated stories told about programmers as opposed to graphical designers, as detailed in section 4.1. Alongside this sits the programmer stereotyping and counter-narratives [2] told around programmer socialisation as above. Further narrative analysis will be conducted into the stories gathered through the study into programmer stereotyping and role archetypes. This may provide insight into how roles are perceived and negotiated within the group, and affected by the shift in view of what constitutes a capable programmer.

5.2 Programmer Humour

As has been discussed throughout the paper humorous stories seemed central to working life in the company and were frequently used to prompt conversational discussion around programming issues. Humour in interaction (see [23]) can be seen to facilitate common understanding and introduce topics for serious discussion; such as the stories of failed street signs and inappropriately answered phone calls. It is also useful for a sense of social cohesion, such as the shared joke about poor web design given previously.

However, although such small stories [12] serve an important interactional purpose, humour can also be misleading:

> CD: We're having a meeting.
> DEV5: Are you?
> CD: Might be.

Although the comment is made in a joking fashion it is hard to be sure whether it is also true (which it turned out it was). This was again responded to in bewilderment by the new staff member. He states in his interview: "They're all really really really nice though. I'm still finding it a little hard to get on with them, a little bit. [. . .] I mean they seem to have a certain entire weird humour that I just don't get."

These humorous stories-in-interaction are being analysed in greater depth using schemes developed by Neil Norrick [23] to further consider how they are used in this context, and to investigate whether they provide insight into the development process.

5.3 War Stories and Metaphor

The previous two sections proposed for analysis directly arose from the ethnographic data. A third area for further analysis into the data instead arises from the literature.

According to Julian Orr [26], as part of his ethnographic study discussed earlier, war stories can "originate in problematic situations and are told or retold in diagnosis when the activity they represent becomes problematic again. . . " ([26] p.126). Further analysis shall thus be made into programmer language [43] for team coordination [28] in the data, specifically in relation war stories and related metaphors within the data collected. It is expected that such stories can be informative about programmer storytelling, taboos and myths [20] about problem cases and possibly provide insight into how such stories are used. Pentland notes that "These are not the grand meta-narratives of progress and modernity; these are the little narratives of "making a good decision" or "designing a good product" What counts as good in these stories emerges from the participants point of view, as do their views of the tools required to enact these stories" ([27] p.7).

It is expected that further analysis into this aspect may reveal whether, and if so how, small stories are used as a knowledge artefact in this context.

6 Discussion

6.1 Limitations and Value

The research presented here is still under analysis so only preliminary observational findings, are presented in this paper. However, there are also general limitations to the approach in general that should also be considered.

As previously stated, narratives cannot be considered in a vacuum and must be considered in relation to the wider situational context and to the interaction-in-context in which they are told. Any account needs to consider this setting, and thus lends itself well to an ethnographic approach. However, in the same way it is limited. The researcher in a narrative-ethnographic approach cannot be considered an unbiased objective observer. Not only does their background and history affect the way they view data but, in a simple sense any ethnographic study features the researcher by their very presence. This can be nicely exemplified by a staff member's joke about hitting a customer with a rubber pipe. It was immediately followed by a smiling comment from the company director saying "remember this is all going down on tape forever."

In addition this leads to the realisation that the study presented is a single study in context, and not an attempt to create generalisable results of findings. This has been neatly discussed in terms of a 'so what factor' [35][36]: "Understanding the reality of practice can, among other things, allow us to prepare newcomers to the field [. . .]; help us to sustain the community and to encourage other communities to flourish; and provide information so that we can recognise what works and what doesn't work" ([35] p.372). The next section attempts to present a future direction for this research and outline how it may still be of value.

6.2 Continuation - Across Methodologies

The company observed was one where members seemed confident, capable, proud of work produced, and strongly bonded through humour and shared stories. The team discussed in this paper are non-dedicated users of Agile Methods, using the agile methodology Scrum [34], and this was borne out by observations.

However they also bear a strong cultural resemblance to dedicated practitioners of the agile methodology XP [3] (although organisational culture "and its interaction with XP practice is complex and not amenable to straightforward categorisation"([32])). The research above is of a company which values communication, simplicity, feedback and courage (the four original values of XP [3]); and where individuals (with the possible exception of the new team member) and the team demonstrate respect, responsibility, faith in themselves and the team as well as a concern for preservation and optimisation of the quality of working life [31].

As briefly stated earlier the company selected was opportunistic, however it provides a good ground for further studies into this area. It is expected that the foundation study which has been conducted here, and the narrative study proposed, will lead to a broader study into the use of stories across a range of software methodologies. The data will potentially form a contrast across the spectrum between 'strict-use' or dedicated agile development practitioners and more traditional (e.g. waterfall) software development approaches, although each case study will stand alone.

It may thus prove interesting to investigate from a comparative sample whether the nature of the stories told varies across methodologies and if so, what this can tell us: What role do stories play? How are social and small stories used in each case? Are they used differently? Are the stories told qualitatively different? Are they encouraged or affected by the processes used? If so, can we learn anything from such differences? Are there implications which emerge?

This approach, although broad at this stage, has clear exploratory potential.

7 Conclusions

This paper has presented and summarised preliminary data from an exploratory narrative-ethnographic field study, presented areas for further narrative analysis, and discussed plans for future development.

The research presented here is broadly concerned with the social aspects of programmer collaboration in systems development, specifically as demonstrable in shared workplace and project-related stories. It is proposed that programmer stories can be explored in greater depth using a narrative research qualitative approach, in order to explore issues around communication, collaboration and culture in software development and investigate how stories are used in practice.

As stated in the previous section, it is expected that this research will support and deepen current research into programmer communication, especially in the context of increasing interest in the value of programmer stories. This work will lead to a broader investigation into how stories are used and told across software development methodologies.

Acknowledgements

Johanna Hunt is a first year DPhil researcher and Associate Tutor at the University of Sussex, and a Research Assistant in Algorithms at the University of Hertfordshire.

This work has been conducted under the supervision of Pablo Romero and Judith Good in the Ideas Lab at the University of Sussex and with the support of the now caffeinated team at Future Platforms.

References

1. Rosio Alvarez and Jacqueline Urla. Tell me a good story: Using narrative analysis to examine information requirements interviews during an erp implementation. *The Data Base for Advances in Information Systems*, 33(1):38–52, 2002.

2. Michael Bamberg and Molly Andrews, editors. *Considering Counter-Narratives: Narrating, resisting, making sense.* Studies in Narrative 4. John Benjamins, Amsterdam, 2004.
3. Kent Beck. *eXtreme Programming explained: Embrace change.* Addison-Wesley, San Francisco, 2000.
4. D M Boje. The storytelling organization: A study of story performance in an office-supply firm. *Administrative Science Quarterly*, 36:106–126, 1991.
5. S. Bryant. Double trouble: Mixing qualitative and quantitative methods in the study of extreme programmers. In *VL/HCC*, pages 55–61, 2004.
6. Alistair Cockburn. *Agile Software Development.* Addison-Wesley, 2001.
7. B. Czarniawska. *A Narrative Approach to Organization Studies*, volume Volume 43 of *Qualitative Research Methods.* Sage, London, 1998.
8. T. DeMarco and T. Lister. *Peopleware: Productive Projects and Teams.* Dorset House Publishing Co. Ltd., New York, 1999.
9. M. Freeman. Culture, narrative, and the poetic construction of selfhood. *Journal of Constructivist Psychology*, 12:99–116, 1999.
10. C Geertz. *Local Knowledge: Further Essays in Interpretive Anthropology.* Basic Books, New York, 1983.
11. A. Georgakopoulou. The other side of the story: towards a narrative analysis of narratives-in-interaction. *Discourse Studies*, 8(2):235–257, 2006.
12. A. Georgakopoulou. Thinking big with small stories in narrative and identity analysis. *Narrative Inquiry*, 16, 2006.
13. Erving Goffman. On cooling the mark out. some aspects of adaption to failure. *Psychiatry*, 15:451–63, 1952.
14. I. Hutchby and R. Woffitt. *Conversation Analysis.* Polity Press, Cambridge, UK, 1998.
15. William Labov. The boundaries of words and their meanings. In Charles-James N. Bailey and Roger W. Shuy, editors, *New Ways of Analyzing Variation in English*, pages 340–373. Georgetown University Press, Washington, D.C., 1973.
16. Charlotte Linde. *Life stories: The creation of coherence.* Oxford University Press, New York, 1993.
17. Charlotte Linde. Narrative and social tacit knowledge. *Journal of Knowledge Management*, 5(2):160–170, 2001.
18. Anthony Lyons and Yoshihisa Kashima. Maintaining stereotypes in communication: Investigating memory biases and coherence seeking in storytelling. *Asian Journal of Social Psychology*, 9:59–71, 2006.
19. S. Marsen, R. Biddle, and J. Noble. Use case analysis with narrative semiotics. In G. Pervan, editor, *Australasian Conference on Information Systems (ACIS2003)*, Perth, Western Australia, 2003.
20. Lindsay Marshall and James Webber. Gotos considered harmful and other programmers' taboos. In A. Blackwell and E. Bilotta, editors, *12th Workshop of the Psychology of Programming Interest Group*, Cozenza, Italy, 2000.
21. Steve McConnell. Ophans preferred, December 22, 1999 1999.
22. Bonnie A. Nardi. Beyond bandwidth: Dimensions of connection in interpersonal communication. *Computer Supported Cooperative Work*, 14:91–130, 2005.
23. Neil R. Norrick. *Conversational Narrative: Storytelling in everyday talk.* Current Issues in Linguistic Theory 203. John Benjamins, Amsterdam, 2000.
24. E. Ochs and O. Solomon. From the outside-in: Practical logic and autism. In R. Edgerton and C. Casey, editors, *A Companion to Psychological Anthropology: Modernity and Psycho-cultural Change.* Blackwell, Oxford, 2005.

25. Elinor Ochs. Narrative. In Teun van Dijk, editor, *Discourse as Structure and Process*, pages 185–207. Sage, London, 1997.
26. J. Orr. *Talking about machines: An ethnography of a modern job*. Cornell U.P, Ithaca, N.Y., 1997.
27. Brian T. Pentland. Narrative methods in collaborative systems research. In R. Sprague Jr., editor, *Thirty-Second Annual Hawaii International Conference on System Sciences (5-8 January 1999)*, volume 1, Maui, Hawaii, 1999. IEEE.
28. M. Petre. Team coordination through externalised mental imagery. *International Journal of Human-Computer Studies*, 61(2):205–218, 2004.
29. James Phillips. Psychopathology and the narrative self. *Philosophy, Psychiatry, & Psychology*, 10(4):313–328, 2004.
30. C. Reissman. *Narrative Analysis*, volume 30 of *Qualitative Research Methods*. Sage, London, 1993.
31. H. Robinson and H. Sharp. Xp culture: Why the twelve practices both are and are not the most significant thing. In *Agile Development Conference (ADC'03)*. IEEE, 2003.
32. H. Robinson and H. Sharp. Organisational culture and xp: three case studies. In *Agile Development Conference (ADC'05)*. IEEE, 2005.
33. Hugh Robinson, Pat Hall, Fiona Hovenden, and Janet Rachel. Postmodern software development. *The Computer Journal*, 41(6), 1998.
34. Ken Schwaber. *Agile Project Management with Scrum*. Microsoft Press, Washington, 2004.
35. Helen Sharp and Hugh Robinson. An ethnographic study of xp practice. *Empirical Software Engineering*, 9(4):353–375, 2004.
36. Helen Sharp, Hugh Robinson, and Mark Woodman. Software engineering: Community and culture. *IEEE Software*, January/February:40–47, 2000.
37. V. Skowronski. Do agile methods marginalize problem solvers? *Computer*, 37(10):118–119, 2004.
38. L Suchman. Making work visible. *Communications of the ACM*, 38(9):56–64, 1995.
39. Felix B. Tan and M Gordon Hunter. Using narrative inquiry in a study of information systems professionals. In *36th Hawaii International Conference on System Sciences (HICSS'03)*, Hawaii, 2002. IEEE.
40. J Van Maanen. *Tales from the Field: On Writing Ethnography*. Chicago University Press, Chicago, 1988.
41. Erica L. Wagner. Interconnecting information systems narrative research: An end-to-end approach for process-oriented field studies. *Global and Organizational Discourse about Information Technology*, pages 419–435, 2002.
42. M. Weinberg, Gerald. *The Psychology of Programming*. Dorset House Publishing Co. Ltd., New York, 1998.
43. Julian Weitzenfeld, Tom Reidl, Charles Chubb, and Jared Freeman. The use of cross-domain language by expert software developers. *Journal of Metaphor and Symbolic Activity*, 7(3-4):185–195, 1992.
44. W F Whyte. *Street Corner Society: The social structure of an Italian slum*. University of Chicago Press, Chicago, 1943.
45. Laurie Williams and Alistair Cockburn. Agile software development: It's all about feedback and change. *IEEE Computer*, pages 39–43, 2003.