

Agile Stories: Agile Systems and Narrative Research (Research Summary)

Johanna Hunt, Pablo Romero, and Judith Good

IDEAs Lab, University of Sussex, Sussex, UK.
{j.m.hunt, pablor, j.good}@sussex.ac.uk

In a period of increasing concern about emphasising collaboration and group communication within software development practice, especially within agile systems development, it is worth pausing to consider the details of communication in practice, and what such consideration may be able to tell us about the people, communities and practices involved in such work.

One feature to be considered is that of conversational storytelling; the day-to-day interactional shared stories that practitioners tell about projects, people and programming. These stories are ‘small stories’, and are commonly neglected as they are not traditionally researched amongst the large concerns of social science or directly quantifiable for computer science research. They are not grand narratives, but small narratives of what makes a project work, how to design a good product, of making some code work, how one person succeeded or failed, how shared humour can make teams or break them, how communication failed, or how code was misinterpreted. Such stories are usable as guides to practice, interaction, and troubleshooting or problem resolution and are not always generalisable as they are dependant on context and teller.

An exploratory ethnographic-style study has been conducted at a small software development company using agile development techniques. Company and personal stories were gathered through observation, field notes and audio recordings of the company at work, as well as through extended narrative interviews, and are currently under analysis. Initial analysis indicates interesting features for further investigation including issues around programmer stereotyping and programmer humour. Recurring stories were told around office space and communication issues, discussions around programmer-graphical designer divides, and humorous stories were told throughout. Humorous stories appeared to be central to working life within the company and seemed to facilitate common understanding and introduce topics for serious discussion. The relaxed conversational practice and open working boundaries were indicative of a group where team members were comfortable and confident in their work and the work of the group.

Further, detailed narrative analysis will be conducted into programmer language within the stories gathered, for team coordination and collaboration, especially in relation to ‘war stories’ and related metaphors within the data collected. These are stories told about problematic situations encountered and faced, and are commonly reused for diagnosis when such situations arise again. The analysis of such stories can be informative about programmer storytelling into taboos and myths about problem cases, provide insight into practitioner identity and practice, as well as possibly support investigation into how such stories are used in practice.

A striking example in initial analysis was given by a new team member, who was observed to be having difficulty settling into his working environment. Relaxed styles of meeting, open plan space, and blurred working boundaries made it hard for the team member to distinguish where he should or could be. In his follow-on narrative interview a few months later it seemed that he was only very slowly adjusting to the group environment. The analysis supported an interpretation indicating a belief that this was in part due to his being a stereotypical programmer. His placement was as a person who, despite technical superiority, found it hard to communicate and to work within what seems a rapid-paced work environment (“*quantity, quality, short space of time*”) where his co-workers are all “*socially intelligent*”. His narration works in opposition to the view of agile systems developers as capable communicators. It is arguable that such a view of self as stereotyped programmer could be potentially disastrous to any team project, as there is a demonstrable lack of sensitivity to unspoken messages and communicative practice.

He provides a ‘war story’ of failed interaction with a client; how he wrote a solution to a problem (“*a giant*”), which was placed as proof of his intelligence and programming skill for solving problems under pressure. He narrates: “*I’ve got to make this deadline so I wrote this amazing task scheduler to the system.*” It was “*very, very, very clever.*” However, it initially caused more problems than it solved as he “*thought [the client] understood the concepts,*” “*I thought they understood from all the conversations I’d had*” but he discovered that “*they didn’t.*” Here he frames his initial failure of implementation in terms of a lack of technical skill on the side of the customer; his code was still good.

PPIG Newsletter – September 2006

However, the above can also be understood in terms of communication failure (“*conversations I’d had*”), rather than technical success in complex individual problem solving. The story is given within a context where the team member is still not fully integrated into the working group, and communication is a key factor given. He understood that in his role: “*when I learn communication I’ll be worth my weight in gold.*” He thus explains and justifies his lack of success, in communication and interaction with the group and with clients, in relation to the programmer stereotype.

This story exemplifies a case where analysis draws attention to stereotyping and communication in software development practice. In this case, the traditional programmer problem-solver role, with pride of artistry in individual creation, appears marginalised within agile practice. The emphasis is on communicative skill and success in interaction rather than on technical ability. Programmers with low social skills may function less well, or potentially be excluded, within an agile practice environment.

The above analysis is one aspect of the research which shall be developed further, as it provides an interesting perspective on agile practice, where communication skills are recognised as just as vital to project success as technical ability. Subsequent research is planned to explore this and related features within agile practice. It is hoped that this will further expand our knowledge and shed insight into communicative practices, and stories told, within the agile communities observed.

Johanna Hunt is a first year DPhil researcher and Associate Tutor in Informatics at the University of Sussex, and a Research Assistant in Algorithms at the University of Hertfordshire. This work is being conducted in the IDEAs Lab at the University of Sussex under the supervision of Pablo Romero and Judith Good.