

**Ethnographies of Programming: Position paper**  
**Dr.Carey Jewitt**  
**c.jewitt@ioe.ac.uk**

I am attempting to understand how the idea of the 'social' and the 'cultural' is conceptualised and realised by social scientists and computer programmers in the use and design of technologies through the process of learning to program. My interest in this is informed by (1) the multidisciplinary context I work within and (2) the focus of my work on technologies and learning.

**(1) The multidisciplinary context I work within**

I work at the London Knowledge Lab, Institute of Education, London: a new specialist centre that brings together computer scientists and social scientists to explore learning and communication more generally. It is nothing new to note the difficulty of these two communities in communicating with each other as well as their desire and interest in doing so.

Despite the interest in working together the ways in which we talked in joint seminars often seemed difficult: concepts often meant entirely different things, there was little common historical ground in the way we used concepts and approached our work. In short, words often turn out to be of limited use in understanding each others' perspectives despite the initial moment of hoped connection. A key aspect of this difficulty seemed to be the way in which we conceptualised the idea of the 'social' and the 'cultural' in the use and design of technology. Many of the people on the 'social sciences side' seemed to think that those on the 'computer science side' simply did not have an idea of the social.

**(2) The focus of my work on technologies and learning**

At the same time as moving into this new work environment I was becoming increasingly aware that my research on the use of technologies in the classroom was limited by a focus on the potentials of the interface without having any sense of how the interface is produced by its source program. I was writing about the semiotics of what I could see on screen as if there were no affordances (limits or constraints) on its design. In a sense I was writing about the screen as if it could be designed in the same way as a bit of paper. I began to realise that aspects of design that I attributed to the designer's choice might be a matter of programming limitation or requirements. My lack of knowledge of all this began to worry me.

**Learning to program as a boundary (crossing) object/process**

A colleague Yishay Mor an academic in Maths and technology who used to be a commercial software designer and programmer offered to teach me basic programming. We set up two linked blogs to briefly map this initial progress and enable us to communicate:

<http://lklprog.blogspot.com/>

<http://careylkl.blogspot.com/>

This has involved us in a range of tasks and discussion that focus on:

- What is involved in learning
- Ways of thinking in terms of coding tasks
- The role imagination
- Coding tasks
- Learning some basic html/java
- The character of programming
- Building a glossary of terms
- Programming practices

At the moment neither of us are 'blogging' as we have reached a stage where I need to start building/programming an essay in html. Through this activity I am beginning to understand what the social can mean within programming and how specific and contextual it is. Through these practices I am developing a more nuanced sense of programming and this is helping me to think also about the relationships between a community and the individual, technology and meaning making, and the social and material affordance of technologies in use.