Probes as a People-Oriented Method

Connor Graham, National University of Singapore, Singapore, and University of Melbourne, Australia

Mark Rouncefield, Lancaster University, UK

ABSTRACT

This paper discusses some methodological aspects of the use of Probes as a people-oriented method. Different kinds of Probes are reviewed and two separate deployments are documented—Technology Probes and Informational Probes—in a care setting. The authors argue that Probes reflect a post-disciplinary era and a shift of attention beyond ‘the social’ to concerns with individual variation, materiality and the visual in technology design. It is suggested that in an era of ‘everyday development’, Probes can play an important role in providing useful insights concerning the product and potentially the process of programming-type work.

Keywords: IS Design Issues, Probes, Qualitative Research, User-Oriented Design, Visual Methods

INTRODUCTION

Grudin (1990) in his historical account of the evolution of interface research points to transformations in focus with regard to principal users, ‘parent’ disciplines, research methods, units of analysis and the precision and generality of studies. Grudin’s argument is indicative of the focus of computing research and development changing as “the interface” has abstracted further and further away from low-level circuitry to being situated in particular places with particular “social” features, and even retreating and disappearing altogether (Weiser, 1991; Norman, 1998). The shift Grudin describes changes the interface problem space to consider the broader, social aspects of computing beyond the hardware, software and even beyond usability. It also suggests a shift in the “principal user” base, the methods and metrics used to investigate these users and what we can draw from such investigations. Thus users have changed in character from highly trained engineers and programmers to, in his terms, “groups of end users” or, in terms pertinent to this journal, ordinary people. In other words he is arguing that there has been a shift of low-level activities such as programming away from ‘the user’ of the final product or interface. In parallel, methods and metrics have shifted from an almost universal adoption of laboratory experiments deploying quantitative measures hoping for mass generalization to more eclectic approaches, including ‘softer’, ethnographic methods that generate findings more particular to people and the settings in...
which they interact. Thus, it seems, the product of programming, the interface, has also shifted to encompass more aspects of people’s lives.

In parallel to the shifts Grudin describes, from the late 1970s and 1980s, ‘programming’ has broadened its audience to include children (e.g. through Logo and Basic – Papert (1980)), young “digital natives” (Prensky, 2001), non-experts and everyday technology users. The development of the digital literacy movement (e.g. http://www.ictliteracy.info/) has placed technology skills at the heart of education programs, where, “…digital fluency requires not just the ability to chat, browse, and interact but also the ability to design, create, and invent with new media…” (Resnick et al., 2009). Most recently, the ready availability and relatively low barrier to entry for Web 2.0 technologies such as web applications, mashups etc. via authoring tools and content management systems (e.g. Drupal - http://drupal.org/) has narrowed the gap between (and perhaps converged) the ‘traditional’ skills required to generate content and the digital skills required to make it available. This creates opportunities to investigate people’s use of technology as they toggle between creation and use, through drawing on the data they automatically generate from ‘the moment’ (e.g. pathways of use ‘through’ the Internet and authoring technologies, emails, text messages, digital photographs and the orchestration of these different technologies) and what they, more reflectively, observe about what they do (e.g. through blogs, annotating and structuring photo streams). In turn, as this kind of expertise spreads and technical hurdles continue to be overcome, this presents interesting opportunities to investigate people’s lives for the purpose of supporting the design of new technologies. Crabtree et al (2006, p. 282), for example, use the “the digital record” or various logs that “detail interaction within a digital environment” to analyse people’s interactions for the purpose of supporting the design of new technologies while Graham et al. (2009) use blogs to examine how life change can be supported through personal reflection.

These shifts are relevant for People-Oriented Programming because, as Goschnick (2009, p. 837) explains, this term focuses on the individual user in three main ways: “as the central focus of a customised software system addressing heterogeneous needs”; “as a self-ethnographer…gather[ing] their own very-specific data (including in the domestic space)” and; “as end-user developers, coming up with their own solutions to match their personal needs, utilising well-engineered software toolkits.”

In this paper we wish to focus on the second aspect of Goschnick’s four-pronged definition of People-Oriented Programming and consider particular methods that leverage and/or make available a whole raft of data on the individual person. One research approach that has been distinct in its treatment of the individual and that has been widely adopted and adapted in the Human-Computer Interaction and Design communities is that of Probes. Graham et al (2007, p. 29) note how: “The notion of a ‘probe’ can refer to a number of things – and all of these versions, or aspects of them, can and have appeared in HCI research.” Boehner et al. (2007, p. 1077), describe the original version, Cultural Probes as: “designed objects, physical packets containing open-ended, provocative and oblique tasks to support early participant engagement with the design process.” Thus Probes, like surgical biopsies, unmanned spacecraft and investigations into criminal activity are deliberately constructed collections of materials and strategies used to explore and discover more; in this case, more about people’s lives (Gaver et al. 1999b; Graham et al., 2007). Our primary aim here then is to argue that Probes as a method can help us to understand what programming might achieve in people’s lives in the broadest sense, as the computer reaches out to more and more aspects of these people’s lives – family, friendship, leisure, politics, citizenship etc.

The popularity of Probes exposes the eclecticism of particular research communities in computing with regard to method (Boehner et al., 2007) but, we believe, is also indicative of a retreat in the wider computing space from
Computer Use Patterns
[www.igi-global.com/article/computer-use-patterns/55730?camid=4v1a](www.igi-global.com/article/computer-use-patterns/55730?camid=4v1a)

The Travel Machine: Combining Information Design/Visualization with Persuasion Design to Change Behavior
[www.igi-global.com/chapter/the-travel-machine/80362?camid=4v1a](www.igi-global.com/chapter/the-travel-machine/80362?camid=4v1a)