Available online at www.sciencedirect.com
ScienceDirect
Int. J. Human-Computer Studies 67 (2009) 437–454
International Journal of Human-Computer Studies
www.elsevier.com/locate/ijhcs

Engineering the social: The role of shared artifacts

Jeni Paay[a], Leon Sterling[a,*], Frank Vetere[b], Steve Howard[b], Anne Boettcher[a]

[a]Department of Computer Science and Software Engineering, University of Melbourne, 111 Barry St. Carlton, Victoria 3058, Australia
[b]Department of Information Systems, University of Melbourne, 111 Barry St. Carlton, Victoria 3058, Australia

Received 30 June 2008; accepted 11 December 2008
Communicated by C. Sierra
Available online 24 December 2008

Abstract

This paper presents a multidisciplinary approach to engineering socio-technical design. The paper addresses technological design for social interactions that are non-instrumental, and thereby sometimes contradictory or surprising and difficult to model. Through cooperative analysis of cultural probe data and development of agent-oriented software engineering (AOSE) models, ethnographers and software engineers participate in conversations around shared artifacts, which facilitate the transition from data collected in a social environment to a socially oriented requirements analysis for informing socio-technical design.

To demonstrate how this transition was made, we present a case study of the process of designing technology to support familial relationships, such as playing, gifting, showing, telling and creating memories. The case study is based on data collected in a cultural probes study that explores the diverse, complex and unpredictable design environment of the home. A multidisciplinary team worked together through a process of conversations around shared artifacts to cooperatively analyze collected data and develop models. These conversations provided the opportunity to view the data from the perspective of alternative disciplines that resulted in the emergence of novel understandings and innovative practice.

The artifacts in the process included returned probe items, scrapbooks, videos of interviews, photographs, family biographies and the AOSE requirements models. When shared between the two communities of practice, some of these artifacts played important roles in mediating discussions of mutual influence between ethnographers and software engineers. The shared artifacts acted as both triggers for conversations and information vessels—providing a variety of interpretable objects enabling both sides to articulate their understandings in different ways and to collaboratively negotiate understandings of the collected data. Analyzing the interdisciplinary exchange provided insight into the identification of bridging elements that allowed 'the social' to permeate the processes of analysis, requirements elicitation and design.

Keywords: Cultural probes; Agent-oriented software engineering; Shared artifacts; Ethnography; Requirements engineering; Social technical environment

1. Introduction

Despite best efforts, contemporary technologies often fail to meet basic human needs and desires (Bell et al., 2005; Christensen, 1997; Haines et al., 2007; Norman, 1999). Recent developments in design processes have ensured technologies are generally accurate, reliable and usable (Sharp et al., 2007). However, meeting these measurable requirements and qualities constitutes only part of what it means to design technology for people. As social beings we often have loftier needs, such as to experience social connection and empathy, to care for others and be cared for, and to share pleasure. These particular types of social requirements cannot be easily reduced to functional specifications for information provision. In existing software development processes, these social requirements are often neglected or trivialized (Sommerville, 2007). We believe it is valuable to match socially oriented user studies with requirements elicitation methods that are able to identify and document social requirements in a form compatible with existing software engineering methods. Technology in social settings will be of increased value if it demonstrably addresses and fulfills the often ephemeral,

*Corresponding author. Tel.: +61 3 8344 1404; fax: +61 3 9348 1184.
E-mail address: leonss@unimelb.edu.au (L. Sterling).

1071-5819/$ - see front matter
doi:10.1016/j.ijhcs.2008.12.002

and hard to measure, felt needs of people in these environments.

The disciplines of Software Engineering (SE) and Human Computer Interaction (HCI) share the goal of effective technology, but differ in their theories, methods, and terminology in interpreting 'effectiveness'. This difference creates a communicative divide, which is accentuated for technological innovation that focuses on socially complex situations. In our research, we used a multi-disciplinary approach to engineering socially oriented software systems. This approach allows us to combine social understanding of technological use in a human context, extracted using ethnographically informed HCI methods, with SE knowledge and experience of modeling user requirements for software design. In bringing together these two Communities of Practice (Wenger, 1998) we are confronted with similar issues of communication and translation faced by HCI research for well over a decade (Constantine et al., 2003; Cunningham and Jones, 2005; Hughes et al., 1994; Kjeldskov et al., 2006; Viller and Sommerville, 2000). These issues concern how to 'bridge the gap' between ethnography and software engineering for the purposes of designing technology.

In this paper, we investigate facilitating interpersonal interactions between individuals with technology in the home, where social activity is not easily conceived in terms of tasks and goals. We are interested in non-instrumental activities, or activities that cannot be easily decomposed into tasks and sub-tasks; and where the purpose is not necessarily to achieve a goal but to participate in a process. This is illustrated by game playing. Rather than specifying the 'rules' and 'interaction style' necessary for winning, we are interested in the mechanisms that facilitate less instrumental outcomes such as 'engagement' and 'social-bonding'. Clearly 'rules' are not orthogonal to 'social-bonding', but addressing one does not engender the other. Our motivation is to support the non-instrumental characteristics, which may be achieved via any one of a myriad of concrete goals.

We acknowledge that non-instrumental activities also occur in the workplace and are often embedded within purposeful tasks. However, the domestic environment provides more acute and intangible instances of them. While it is true that purposeful work gets done at home, it is the activities that remain when work is abstracted out of family life that we find particularly interesting.

Social requirements obtained through ethnographically informed HCI methods are generally not in an appropriate form for simply feeding into traditional software analysis methods. In our case, cultural probes were used to provide access to people's daily interactions in the domestic setting. These interactions are difficult to study using traditional empirical techniques such as questionnaires, focus groups and participant observation. The data gathered using probes is fragmentary and unstructured, and in the absence of any proven method, the process of translation from probe data to the abstract generalization required in software design is not an easy one.

In our project, we created software requirements models with the agent-oriented software engineering (AOSE) methodology ROADMAP (Juan et al., 2002; Kuan et al., 2005) from the cultural probe findings. We were particularly interested in testing ROADMAP's ability to represent non-instrumental social requirements. This is because its notation extends beyond functional goals, used in traditional software engineering methodologies, to include a special type of goal called a *quality goal*. Quality goals are essentially non-functional and are designed to encapsulate social aspects of the context into the software requirements model, thus providing a mechanism to carry social aspects through to the implementation phase. Identifying quality goals became an important part of the requirements elicitation phase for capturing social requirements from the probe data.

The process of translation was enacted in team meetings, where members of a multidisciplinary team worked together to analyze probe returns and identify quality goals, while creating and maintaining their own representations of understanding. Because the two communities of practice involved came to the table with different values, practices, orientations to technology, and commitments to the process, they achieved this analysis through exchanging thoughts, interpretations and understandings in a series of conversations mediated by a collection of artifacts. These artifacts had different purposes, qualities and affiliations, and were used to cross and negotiate boundaries (Lee, 2007; Star and Griesemer, 1989) between the two groups working within an 'artefact ecology' (Vyas and Dix, 2007) consisting of different digital and physical artifacts, the members of the multidisciplinary team, and their work practices and values. The role that the shared artifacts played facilitated both communication between disciplines and the embodiment of interactions and work coordination that such artifacts play in meetings generally. Coordination was primarily achieved by conversations around artifacts, and it was these conversations that team members found the most enlightening part of the exchange. The conversations enriched their own understandings of the design situation as unanticipated viewpoints emerged; exploiting the complementarities of the different value sets and approaches to design of the two communities of practice.

Artifacts are a powerful resource for analysis, they "tell a story to the extent that they invoke stories" (Ramduny-Ellis et al., 2005, p. 77). They can represent the understandings of one individual and also be used to mediate and negotiate work in collaborative settings (Vyas and Dix, 2007). By analyzing the attributes of the artifacts that made them function as useful shared objects, or not, we can better understand the role that shared artifacts played in the process of translation of ethnographic understanding to abstracted design model.

This paper sits in the territory of the relationship between ethnography and software engineering, and we ask the question: *what is the role of shared artifacts in supporting multidisciplinary teams in engineering the social?*

To do this we use our case study to explore the roles that artifacts played in the process of analysis, and the attributes and affordances of those shared artifacts that expedited this.

This paper is structured as follows. Firstly, we present background information about past research into social analyses in HCI and requirements engineering, including the value of an agent-oriented approach to this. Secondly, we look at relevant aspects for socially oriented requirements engineering for the domestic environment. Thirdly we present our approach to the process of analysis using a multidisciplinary team and shared artifacts. Fourthly, we present our case study. We then give the details of our analysis of field data and creation of requirements diagrams to illustrate engineering of the social. Finally, we discuss the insights gained from examining the role of shared artifacts in this process, and then conclude on our findings on the role of shared artifacts.

2. Background

2.1. Social analyses and requirements engineering

Research into bridging the gap between ethnography and software engineering has taken place in the HCI and SE communities for well over a decade. At the heart of the matter of bridging this gap is the difficulty in communicating ethnographic results to the software engineers for the purpose of software design (Hughes et al., 1997). As early as 1993, Goguen (1993) stated that an important research problem at the time was how to integrate ethnographically inspired understandings of the social issues of systems needs with conventional requirements engineering methods and notations. The problem is finding a suitable mechanism for the transference of knowledge between these two fundamentally different disciplines. Both use different terminology, and different methodologies. Ethnography deals in "the particular", and software design in "the abstract" (Viller and Sommerville, 1999). The outputs from ethnography are not related to the kinds of representations required by SE (Hughes et al., 1995b). Ethnographers work with notes, reports and transcriptions based on rigorous observations. They produce results in a discursive form to provide the reader with a very detailed rich and concrete understanding of the everyday practical aspects of human interaction in context. On the other hand, software designers and engineers create and manipulate formal graphical abstractions. They spend their time immersed in notations and descriptions as techniques to simplify the complexity of a situation and to represent critical features of it. Ethnographers avoid judgements; designers make them (Paay, 2008). Ethnographers take an analytical role, whereby they gather and interpret data. On the other hand, software engineers synthesize data through the use of abstract models of situations (Button and Dourish, 1996; Hughes et al., 1995a).

The purpose of requirements engineering is to determine what properties a system should have in order to succeed (Goguen and Linde, 1993). However, the attitude of requirements engineers in the early 1990s was that the social, cultural and political aspects of work either fell outside of their scope and should be ignored in requirements gathering, or that those issues were simply inherently embedded in the requirements that they gathered, because users are situated in social worlds. In the late 1980s and early 1990s pioneering work by sociologists, such as, Lucy Suchman, John Hughes, Richard Harper, Christian Heath and Paul Luff, inspired the use of ethnography for understanding the social aspects of work processes and informing system design (Hughes et al., 1995a). This was motivated by a growing need to design for complex real world situations, and began with the belief that methods from the social sciences, such as ethnography and ethnomethodology could provide a means for better understanding the contextual issues of technology use. However, researchers struggled with the challenge of how to translate the insights provided by ethnography into the activity and representations of software design.

Toward the end of the 1990s, systematic approaches to social analyses were being explored by researchers with the aim of integrating this information into the design process including the use of collaborative analysis meetings between ethnographers and software engineers and multidisciplinary teams participating in both the fieldwork and the design process (Diggins and Tolmie, 2003; Hughes et al., 1995a; Viller and Sommerville, 1999). Researchers also explored presenting the ethnographic record in more software engineer friendly representations such as extended UML notation (Viller and Sommerville, 1999), pictorial stories, drawings, data models, analogies and metaphors (Millen, 2000), dedicated shared software packages (Diggins and Tolmie, 2003), pattern languages (Martin and Sommerville, 2004), and design documentaries (Raijmakers et al., 2006). However, despite past research efforts trying to bridge the gap between ethnography and software design this still remains a matter of concern to researchers (Rahwan et al., 2006; Schraefel et al., 2004; Sutcliffe, 2005; Walenstein, 2003; Wiltshire, 2003).

In contrast, our approach does not have a representation that is *a priori* common. Our approach is based on the multidisciplinary team, where a cooperative shaping of the findings is done in team meetings through conversations. These conversations were connected to a variety of shared artifacts that played several roles. The artifacts are shared in as much as they are given over for perusal and consumption, but they 'belong' to one or other group. Thus our contribution is in understanding the role these artifacts played in facilitating this translation, and the ways in which they shaped, evolved and promoted the social aspects of interaction into the design. The use of shared artifacts in mediating between HCI and SE processes is not unusual (e.g., Constantine and Lockwood, 2003; de Paula et al., 2005; Rahwan et al., 2006; Sousa and Furtado, 2003;

Vyas et al., 2006). A team working with shared artifacts can become aware of more interpretable possibilities as a result of using that artifact (Ramduny-Ellis et al., 2005). Artifacts are also at the centre of several conceptual and theoretical frameworks widely used to understand the context of users' work and non-work, such as, activity theory, distributed cognition and structuration theory (Vyas and Dix, 2007). 'Artefact ecologies' are used to study the role of artifacts in supporting meeting practices with distant access. Vyas and Dix (2007) define three key attributes of artifacts: the content that they embody, their significance to different people, or groups of people, and their disposition in the environment. These attributes can be used to describe the relationship of artifacts to the meeting practices that they are supporting.

We are interested in how the shared artifacts were used by the multidisciplinary team. By analyzing the role of artifacts in team meetings, including their affordances and attributes, we are looking for shared practices and their contribution to the success of the conversations that took place in meetings. We want to understand how shared artifacts bridge between two ways of thinking—HCI and SE—and how they give traction to incorporating 'the social' into the engineering process.

2.2. The value of an agent-oriented approach

Most requirements engineering research to date has focused on the workplace and on examination of existing systems. Valuable research has been done to integrate ethnographic results into software design process (Kjeldskov et al., 2006; Viller and Sommerville, 2000). However, this research concentrates on object-oriented software engineering, which does not explicitly have notations or constructs for including information pertinent to social contexts into the system design. Social issues were recognized as being a key component of requirements engineering from the early 1990s (Goguen, 1993) with focus on general issues rather than specific mechanisms.

The i* framework (Yu, 1995) was an attempt to approach requirements engineering from a broader perspective. It utilizes Strategic Dependency models to describe the network of social dependencies among actors in the system. The i* framework led to approaches for explicitly handling qualitative system aspects using quality requirements, often called non-functional requirements (NFRs) (Chung et al., 2000).

In another direction, much research has been done on goal-oriented requirements engineering (GORE), of which KAOS is a well-known example, or on goal and scenario modeling (van Lamsweerde and Letier, 2004). The GORE methodology does not focus on NFRs. It emphasizes a top-down approach rather than abstracting programming constructs up to the requirements level, and emphasis is clearly on the system, for example explaining requirements to stakeholders rather than eliciting requirements from them. GORE concentrates on using goals for requirements elicitation and analysis, requiring the cooperation of active agents, in this context meaning simply active components, to achieve goals; however, it does not adhere to the agent paradigm.

The agent-oriented software engineering (AOSE) paradigm is promising for socially oriented software systems development because the characteristics of agents can be compared to humans. Agents may exhibit human-like behaviors such as autonomy, flexibility, intelligence, learning, and dynamic adaptability to the surrounding environment, increasing their suitability for socially oriented software facilitating people's social interactions. An early and well-cited AOSE methodology is Gaia (Wooldridge et al., 2000), which took an organizational view and describes systems in terms of goals and roles. Within the realm of AOSE methodologies, the TROPOS methodology is well known for being requirements-driven and strong in the early requirements phase (Castro et al., 2001). TROPOS defines systems in terms of autonomous, intentional and social software actors, and incorporates concepts from i*. TROPOS is referred to as requirements-driven because it is based on concepts used during early requirements analysis, such as actor, social dependencies among actors, etc. It documents requirements in a way compatible with a business environment and to facilitate software engineering tasks from design down to implementation. TROPOS is a rather formal methodology, in which modeling quickly becomes complex and which is not particularly suitable for capturing ambiguity, uncertainty or play. However, the ability of TROPOS to capture NFRs and dependencies between NFRs make it a possible candidate methodology for attempting socially oriented requirements engineering.

Role-Oriented Analysis and Design for Multi-Agent Programming (ROADMAP) is an AOSE methodology that started as an extension of Gaia but became a methodology in its own right (Juan et al., 2002). Analysis and design in ROADMAP differs from the above-mentioned methodologies because it enables extensive, detailed and fine-grained capture of the different roles required in a software system, as well as the envisioned goals. ROADMAP clearly separates analysis and design and is independent of design architecture and agent platform. Furthermore, ROADMAP exhibits strong flexibility, which enables uncertainty and ambiguity to be captured during the requirements elicitation phase. For example, most requirements engineers believe that a very important elicitation task is to establish system boundaries (Nuseibeh and Easterbrook, 2000). ROADMAP, in contrast, intentionally leaves open decisions about system boundaries during requirements analysis, postponing them as much as possible until later. The simplicity of the role and goal model diagrams, with which requirements analyses are depicted, enable presentations to potentially non-technical audiences, such as interaction designers and cultural probe participants, which are easily understood and which generate lively discussion. Using ROADMAP we were able to elicit a set of roles, goals and quality goal requirements that could usefully and usably be passed to an AOSE system design tool, the Prometheus design tool (PDT) (Thangarajah et al., 2005), built on the Prometheus methodology (Padgham and Winikoff, 2004).

Quality goals are analogous to NFR's in TROPOS, but emphasize a more active searching for qualities of the interaction that are distinct from functional goals. They are there to represent quality requirements, of the sort that are found in social environments, and more abundantly in home environments. Quality goals are less amenable to decomposition than the clearly hierarchical functional goals of a system. In addition to this, quality goals actively promote ambiguity. They are intentionally ambiguous, and this ambiguity is not presented as a problem to be solved. Instead they claim the significance of ambiguous elements in social systems, such as flirtatiousness, fun and play, and carry these through to the design phase in a system independent form.

Eliciting requirements from probe data involves working in a milieu in which capturing concepts accurately but flexibly at a high level, without losing the liveliness and vitality of those concepts by specifying all details, is considered essential. From our experience, ROADMAP appeared a highly suitable SE methodology for achieving this because it enabled the elicitation and representation of high-level non-functional requirements through its use of quality goals. Roles in ROADMAP can be used to represent real humans, or may be consolidated into agents to form part of the multi-agent software system. We used roles to represent those responsible for a particular set of goals. The analogy to humans, provided by using roles and agents, made ROADMAP models a useful mediator when communicating software engineering designs to our multi-disciplinary team. Discussions were focused around interaction issues rather than on more traditional software engineering issues such as the problem domain. We used ROADMAP to bridge between the results of our cultural probe studies to the design of an appropriate software device for facilitating social connections and relationships.

3. Socially oriented requirements engineering

3.1. Domestic technologies and social interaction

The disciplines of HCI and SE have a plethora of methods and techniques for understanding, analyzing and designing computing for the work domain. Now that computers are becoming increasingly interwoven into the domestic setting, the question has arisen as to whether the understandings and methodologies developed specifically for the workplace are also applicable and appropriate when designing technology for the home (Crabtree, 2004). It is true that when information and communication technologies began diffusing into the home they did so mostly as extensions of our places of work, but this is changing. Technologies are now emerging which allow people to manage busy social and work lives as well as supporting participation in the richness of family life (Harper et al., 2008). The design of these technologies is supported by a decade of investigations into understanding human interaction with domestic technology (Crabtree et al., 2002; Hindus, 1999; O'Brien and Rodden, 1997; Venkatesh, 1996). Recently, HCI and Computer Supported Cooperative Work (CSCW) have begun to explore the home as a distinctive design space, with new challenges that are continually emerging (Bell and Dourish, 2007; Sengers and Gaver, 2006).

Domestic technology is not a new concept—domestic technologies such as fridges, microwaves, telephones and televisions have played an integral part in our daily lives for many years (Frohlich and Kraut, 2003). The introduction of the desktop computer into the home has proven itself to be both a time saving and time using appliance. Where at first it was used for supporting work-related activities that took place in the home, such as keeping budgets, managing appointments and correspondence, home computing is increasingly about socializing and leisure rather than work.

Designing for fostering social interactions in the domestic setting requires new approaches and methods to better understand how to design for the 'ludic' aspects of everyday life such as pleasure, intimacy, creativity and fun (Crabtree et al., 2005; Gaver, 2001; Gaver et al., 2004; Terrenghi et al., 2007). The goal of many family activities can be as diffuse as passing the time, where the family members themselves would find it difficult to explain what had been produced during such an activity (Howard et al., 2007). And yet this is the way in which families define themselves and these are some of the strongest memories that they carry forward with them. New domestic technologies need to foster and support the social interactions that make us essentially human, such as being part of a family, staying connected to each other, caring for one another and growing old in safety and comfort (Harper et al., 2008). These are the qualities that make a house a home (Zafiroglu and Chang, 2007).

It is the social interactions in the domestic setting that interest us in this project. The non-functional aspects of the goals of family activities presents the kind of case that will challenge conventional requirements engineering modeling techniques and allow us to test the ability of the ROADMAP quality goals to represent aspects of everyday life such as pleasure, intimacy, creativity and fun, and to do so with a richness and detailed account that makes the resulting design appropriate for fostering these social interactions.

3.2. Understanding domestic settings and cultural probes

There are well-established theories, models and methodologies in HCI for understanding technology in the work setting (Beyer and Holtzblatt, 1998; Cheverst et al., 2003; Halverson et al., 2004; Neale et al., 2004; Suchman, 1987)—but as discussed above, homes are not necessarily the same as workplaces (Harper, 2003; Hindus, 1999).

Designing technology for the domestic setting involves first studying and analyzing the specific situation and context in which that technology will be placed. To understand domestic settings, many of the analytical techniques devised for studying workplaces are not sufficiently flexible to account for the unexpected, the contradictory or the surprising. The home environment requires a different approach (Hemmings et al., 2002). Despite a decade of interest in the domain, there is no well-defined set of methods for understanding the domestic setting (Cheverst et al., 2003).

Just as ethnographic methods became widely used to assist with requirements elicitation in the workplace as the first step toward creating new technologies (Crabtree, 2000; Hemmings et al., 2002; Hughes et al., 1994; Iqbal et al., 2005a; Viller and Sommerville, 2000), ethnographically informed field study methods are being adopted for gaining insight into the domestic setting. Ethnography is able to provide a rich and concrete portrayal of a domestic environment for the purpose of designing technology (Cheverst et al., 2003; Tolmie et al., 2003). This detailed study of context of the lived experience helps us to design interactive systems for that context (Vyas and Dix, 2007). It provides analyses of the socially organized character of settings as experienced by the people in that setting, bringing a social perspective to system design (Cheverst et al., 2003). Some of the different methods used in understanding the domestic setting include: technology biographies constructed by the researchers and users in the home (Blythe et al., 2002); analyzing the routine communications that happen in the home (Crabtree, 2004); analysis of informational artifacts in the home such as calendars, paper notes, to-do lists (Taylor and Swan, 2005); studying extreme users, such as those with Asperger Syndrome (Vetere and Feltham, 2007); and distributing 'cultural probe' packs (Gaver, 2001).

Due to their flexibility, the cultural probe approach has been adapted and appropriated for a divergent range of purposes, in a variety of different technology projects (Boehner et al., 2007; Graham et al., 2007). These projects include: understanding care facilities for elderly people (Gaver et al., 1999); informing design for a sensitive health care setting (Cheverst et al., 2003); studying intimacy in couples (Kjeldskov et al., 2005); inspiring design for creating new technologies supporting practical needs and playful desires in the home (Hutchinson et al., 2003); and as a data collection method to create structured user needs for a design process in the domestic setting (Haines et al., 2007).

In this project, we explored the use of cultural probes in understanding a particular aspect of home life, that of intergenerational play. We were interested in taking account of social and environmental factors in our design of technology for that particular social interaction. To do this we needed access to the rich mix of perspectives and insights to be able to begin to imagine and sketch out different technological possibilities (Harper et al., 2008). Cultural probe data gave us access to a rich and detailed account of the present setting, as well as inspiring ideas for future design. It provided us with access to the social aspects of interactions in the home. The cultural probe was, after all, designed to garner an understanding of the playful character of human life and the multifaceted ways people "explore, wonder, love, worship, and waste time" (Gaver, 2001).

3.3. Requirements elicitation and the home

In traditional systems design methodologies, for example the object-oriented software engineering approach (OOSE), it is the structured understanding of user needs derived from the requirements elicitation phase that drives the design phase of technology development. Recent SE development methods, such as Iterative Design (Larman and Basili, 2003), Agile Development (Martin, 2003) and Extreme programming (Beck, 2000), have challenged the sequentiality of that paradigm, and even the role that users play in the process, but to date have focused on design for workplace environments and defining functional requirements. The agent paradigm and the field of agent-oriented software engineering, is focused on producing software systems that are flexible, autonomous, situated, and can dynamically adapt to (or learn from) their changing context (Wooldridge, 1999). These attributes of AOSE appear to make it more suitable than other SE methodologies for developing technology to facilitate non-work-related, non-instrumental interpersonal interactions in the home.

This posits the question of whether requirements elicitation, in the traditional sense of the word, is appropriate for defining social interactions. As a consequence of this, interaction designers have moved away from structured design methods, to an inspirational design approach (Blythe et al., 2002; Benford et al., 2005; Gaver et al., 2004; Hutchinson et al., 2003; Raijmakers et al., 2006). Inspirational design methods, based on probe data (i.e., Benford et al., 2005; Gaver et al., 2004; Hutchinson et al., 2003) promote a flexible approach for assimilating information about users into the design process. This approach relies on the creation of prototypes or technology probes to elicit additional user requirements to continually and iteratively refine the design in response to the way they are appropriated. This approach is effective when creating small-scale innovative technology for novel situations, but more complex and large-scale systems require detailed and structured engineering to ensure all requirements have been accounted for. Hence we are back to the situation of trying to find an elicitation method that provides access to non-instrumental social interactions.

Examples of where ethnographically based methods have proven useful in the requirements elicitation phase of software engineering for production of technology are numerous (e.g., Crabtree et al., 2000; Crabtree, 2004; Cunningham and Jones, 2005; Hughes et al., 1994, 1997; Iqbal et al., 2005a, b; Kjeldskov et al., 2006; Neto et al., 2005; Ormerod et al., 2003; Simonsen, 1997; Simonsen and Kensing, 1998; Surendra, 2008; Viller and Sommerville, 2000; Vyas et al., 2006). Less numerous are cases using ethnographically informed methods to elicit requirements for the design of technology specifically for the home. Those who have used probes did gain a sufficient understanding of the lives and family experiences of their participants to create innovative and playful designs (e.g., Arnold, 2004; Gaver et al., 2004; Haines et al., 2007; Hutchinson et al., 2003; Kjeldskov et al., 2005; Vetere and Feltham, 2007; Vetere et al., 2009). These cases where probes have been used for eliciting requirements show the results that can be achieved in terms of innovative domestic technologies, but do not bring us closer to understanding the process of transition from analysis to design.

Probe data was never intended for requirements elicitation (Gaver et al., 2004). They were intended to inspire design rather than specify requirements. An ongoing dialogue and dialectic between participants and researchers is central to probes as a method (Gaver et al., 1999, 2004; Hemmings et al., 2002). One aspect of the dialectic is the need for researchers to interpret, based on their own experiences, the cultural probe data in order to understand it (Gaver et al., 2004). Probes do not provide an explicit method for transforming fieldwork insights into technical applications (Rouncefield et al., 2003), nor is it easy to trace from designs back to probes (Gaver et al., 2004). Nevertheless, probe data is being appropriated for use in design, and it is this translation from field data to eliciting requirements for software engineering that we wanted to examine, through a process of involving both ethnographers and software engineers in the ongoing dialogue to understand the probe data in light of their divergent experiences. In the practice of software engineering, the modeling scheme used in requirements engineering should match the type of data gathered (Nuseibeh and Easterbrook, 2000). In our case we wanted an analysis methodology able to capture social concepts, e.g. playing, gifting, showing, telling and creating memories. We believe that the ROADMAP AOSE methodology was able to do this.

4. The Magic Box case study

4.1. Cultural probes: investigating intergenerational play

The aim of the cultural probe research was to understand playful interaction as a means toward designing technologies to support intergenerational play across a distance. This was motivated by the increased physical separation of grandparents and grandchildren, both geographically and in terms of time zones and schedules. There are many documented benefits from increased intergenerational contact, yet current technologies, such as telephone and e-mail, are not sufficient to adequately support this contact across a distance.

The first stage of the project explored intergenerational play by observing grandparent–grandchild playgroups to provide some theories about their interactions. The second stage involved data collection with six families in Melbourne, Australia, using cultural probes and interviews. Each family had at least one grandparent and two children aged between 2 and 10 years, as well as parents. The grandparents lived in a separate household and had at least weekly contact with their grandchildren. One family consisted of two grandparents and six children across three households, while another family consisted of two children and two separate grandparents, one maternal and one paternal, therefore again distributed across three households. Each family participated for a 2-week period, during which time three interviews took place: at the beginning, after 1 week, and at the end. Initially the families were presented with cultural probe packs. The cultural probe packs contained the following materials: scrapbooks; postage items including boxes, envelopes and tubes; stationary items such as colored pens, pencils and crayons, stickers, scissors, stencils; labels with catchphrases to be continued, such as: "I wish we…" and "I love it when we…", a Polaroid camera and film. After one week, the postage items were withdrawn and replaced by the Magic

Fig. 1. The Magic Box is a cultural probe that carries objects.

Box, a communication and exchange system, illustrated in Fig. 1.

The Magic Box was a colorful cardboard storage box in which to place items designated for the distributed family. One Magic Box was supplied per household. Participants were encouraged to place things reflecting their relationship into the box. For a period of 1 week, the Magic Box was picked up from the front doorstep each night by the magic fairy (a researcher). The Magic Box contents were photographed during the exchange each night. The grandparents' box was transported to the grandchildren's house and vice versa, where they were opened each morning. The families were encouraged to use a scrapbook to record use of the cultural probe items and the Magic Box. These scrapbooks were collected in the final interview. All interviews were video taped.

The first stage of the project found that intergenerational interactions tend to be short and episodic, incidental and playful, rather than structured around more formal games. As to be expected from cultural probe results, the findings from stage two were of a fragmentary and highly informal nature. The rich materials from participants were discussed and interpreted in multidisciplinary meetings where both the ethnographers and software engineers participated. Of necessity, interpretation of ambiguity occurred through the researchers applying their own experiences to the situation, although analysis of interviews and scrapbooks sometimes clarified issues first. Participants' motivations were discussed. Activities were abstracted and categorized into types, such as *game*, *instruction*, *ritual*, *gifting*. The data analysis revealed *play*, *gifting* and *storytelling* as the most important types of intergenerational interaction. While exchanging postal items generally didn't provoke much interest in the families, the Magic Box exchange was strikingly different, provoking creative play.

Grandparents tended to use themes, trying to initiate a dynamic conversation with their grandchildren. For example, stories were told about how the grandparents had migrated to Australia. Often photos of grandparents and other family members as children were sent in the box, indicating a desire to tell stories about family history. Some grandparents were strong in informing and teaching, writing letters and notes each day about how to carry out some task. Alternatively, game playing was encouraged: "Can you find Grandma in the photo?" The children tended to send items from their daily lives, reflecting the here and now of play in contrast to the grandparents' reflectivity. Children sent pictures they had drawn, photos taken. Some sent toys, or families of toys, such as horses, for the grandparents to look after, reflecting their conception of the grandparents as carers.

Parents utilized the convenience of the Magic Box exchange to send items such as magazines to the grandparent household. Jokes were also played, with one family father encouraging his children to send cow dung in a plastic bag as part of a vivid description of a day trip to a farm. As a gift, almost all families included food in the Magic Box, in particular sweets as a gift on the last night of the exchange. One family gifted and invited game playing by exchanging jigsaw puzzles. Misunderstandings also occurred, with items being sent back by mistake or in rejection.

The scrapbooks were often used as a diary. Some grandchildren pasted in the letters received from the grandparents. In other families, the scrapbooks were played with when the families got together, with the grandchildren contributing to the grandparent's scrapbook using stickers, drawings and photos.

4.2. Software engineering: eliciting requirements with ROADMAP

ROADMAP was used as the AOSE methodology for requirements elicitation because of its ability to represent very high level requirements, its use of roles, goals and in particular quality goals, and its ability to be understood by members of a multidisciplinary team.

Analysis and design in ROADMAP proceeds by developing a series of models. This project used role and goal models from the current version of ROADMAP as features. These features were chosen because they offer capacities at the requirements analysis level particularly useful for this project. The role models capture a lot of detail, which is an advantage when eliciting requirements and initiating discussions with clients about responsibilities and constraints of roles and systems. System requirements and quality constraints are captured independently of later decisions concerning design and architecture. This provides high-level abstractions and flexibility of design that is important for people to understand the models. ROADMAP also offers scalability, enabling various levels of abstraction to hide complexity or focus on the detail of a particular section, and ease of use. The actual work was done using the ROADMAP case tool, REBEL (Kuan et al., 2005).

Role and goal models in ROADMAP are the application specific models for the analysis phase of the engineering process (see Fig. 2). Goal models represent the systems goals in a loosely hierarchical fashion. Quality goals may be attached to regular goals and constrain how that goal should be fulfilled. Quality goals reflect the intangible goals of a system, like *privacy*, *risk taking*, *timeliness*. Roles are attached to goals, representing those responsible for achieving that goal. Role models consist of a name, a description, a list of responsibilities and a list of constraints. The concept of roles facilitates requirements elicitation in social situations because humans, like agents, take on one or more roles within a particular organizational unit and therefore become responsible for a set of goals.

Goal model diagrams consist of groupings of goals, which may be hierarchical. Goals may be qualified or constrained by quality goals that describe how to fulfill a goal (e.g. *timely*). Quality goals are less concrete, more

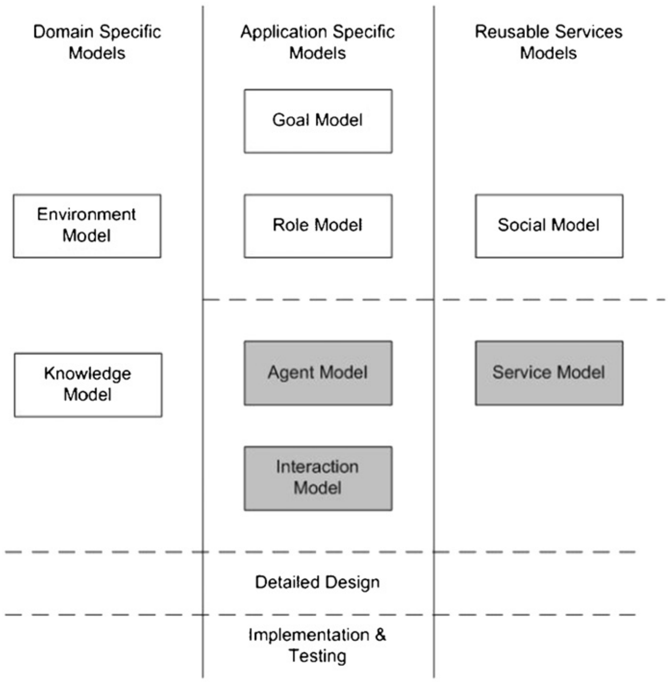

Fig. 2. ROADMAP models (Kuan et al., 2005).

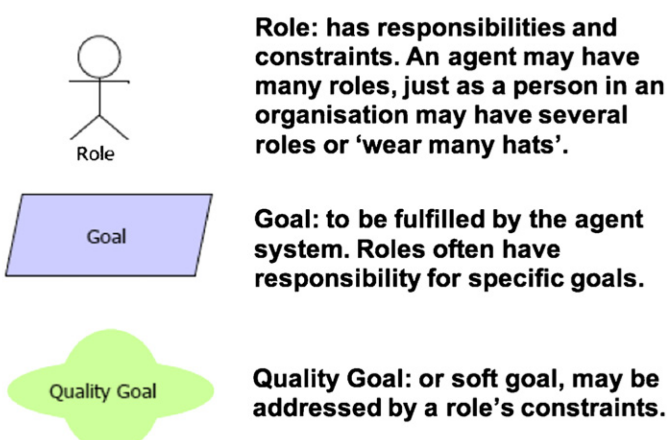

Fig. 3. REBEL requirements analysis notation.

intangible, enabling the capture of subjectivity, such as emotions. Roles are similar to the roles humans take on in life, and further down the track agents, comparable to humans, may take on many roles, or wear many hats. Roles have associated responsibilities, reflecting the tasks or goals they are responsible for, and constraints on how those goals may be fulfilled. In the goal model diagrams, roles are associated with particular goals. The responsibilities of a role should reflect the goals they are attached to, and the constraints should reflect any quality goals that qualify those relevant goals.

The notation for the REBEL tool is shown in Fig. 3. Roles are stick figures, goals are parallelograms, and quality goals are clouds. Arcs or lines are used as connectors in the composite diagrams.

5. Engineering the social: collaborative analysis

The multidisciplinary team went through a process of consultations and conversations in respect to unpacking the ethnographic data that was collected, and the identification of goals and roles within that data. This process involved the set of steps, detailed below, of understanding the probe data, making requirements models, and presenting those models back to the group and refining them in response to feedback. Artifacts played a central role in these meetings, in knowledge transition, in coordination of cooperative work, in coordination of communication, as repositories of analytical outcomes, and as triggers for conversations and differing interpretations. In this section, we describe these steps in our analysis process using the Magic Box case study field data and the ROADMAP methodology to model goals, roles and quality goals from data. At each step we examined the artifacts that motivated the conversations between the different communities of practice and influenced the outcomes of that step. We identified the affordances of each artifact, its attributes, and its contribution in mediating team meetings.

5.1. Understanding the data

In the beginning, it was the software engineers, creating the ROADMAP models, who drove the collaboration by attending the regular weekly meetings of the ethnographers of the Magic Box study. These ethnographers were presenting the outcomes of the cultural probe research back to their colleagues. These meetings gave the software engineers the opportunity to participate in discussions of the cultural probe returns, as well as becoming familiar with the various forms of data being collected, that they would need to understand to begin the modeling process. They soon became part of the team and wherever practical, the software engineers were actually sent out to participate in the data gathering, for example, interviews with participants. This involvement of software engineers in the ethnographic data collection process as well as analysis of the data within multidisciplinary teams are both recommended methods for bridging the gap between HCI and SE (Diggins and Tolmie, 2003; Hughes et al., 1995a; Viller and Sommerville, 1999).

During the initial analysis of the cultural probe data, all team members contributed to interpreting the activities observed between grandparents and grandchildren by referring to and comparing with their personal situations. All opinions were entered into the dialogue and this helped to build an ongoing relationship in the team. Coincidentally, all team members had children, most aged 2–10 years, and most were also able to contribute knowledge of other cultures.

In the field, the ethnographers worked in pairs, each pair becoming familiar with the interactions and exchanges of a single family. At each weekly meeting the pairs of researchers would present back to the group the story of their families. This would include a discussion of the items exchanged by the family using the Magic Boxes. These exchanges were illustrated to the team using digital photographs of the content of the boxes taken during the nightly changeover. The families were interviewed three times during the 2-week study and videos of these interviews were watched by the team after each event, discussed, and used to cooperatively build a biography/profile for each family. These biographies were maintained in a PowerPoint file, and were modified regularly as a team, remaining true to each family and their own stories, rather than generalizing across families. Interview summaries were also built up by the group on a whiteboard after each interview, and recorded in a Word document. The few postage items that were returned were brought to the table for discussion, and at the end of the 2-week study each family handed over their scrapbooks to the researchers, and these were made available to the team. At the end of the entire study of all six families, the insights gained from the experience as a whole was collated and added to the PowerPoint file.

In their desire to inform innovative design, the ethnographers were looking in the data for clues about how these people related to each other and how technology might support that. Many of the objects collected were not able to stand alone (e.g., a scribble in a scrapbook), but required the interpretation of the researcher working with that family (i.e., this is the contribution of a very young child). In this way, some objects came to the table pre-digested, in other cases, the team would discuss to reach a consensus of what they might mean. The ethnographers were deliberately searching for the ephemeral instances, things that were remarkable, surprises, interesting events, things that were hard to describe, and events that caught their imaginations. They played with ideas that they considered would be difficult to engineer using current communication technologies. They were looking for items that would help them make sense of the domestic space, and things that were playful, fluid and interpretive. They intentionally tried not to summarize because they wanted to maintain the richness and complexity of family relationships. For them the outcomes from the probe study were the detailed stories about each family.

On the other hand, the software engineers were continually scanning the conversations for general concepts that could meaningfully lead to a design. They needed to abstract, to create a blueprint for a product line of technology that was not just implementable, but had general appeal and application in the community, because for software engineering, commercial success indicates innovation (Davila et al., 2006). At the same time, in line with their aim to add a social dimension to SE modeling, they were seeking confirmation from the ethnographers that their identified quality goals were capturing the essence of the social interactions being modeled.

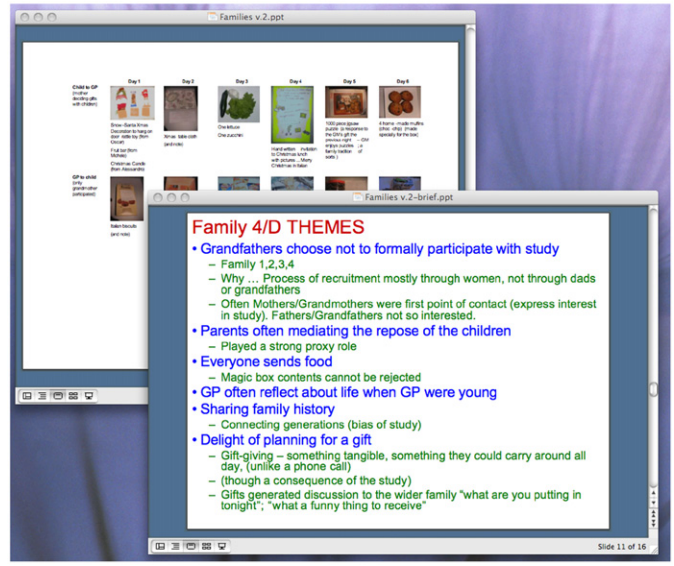

Fig. 4. The PowerPoint file of ethnographic analyses projected.

The most influential shared artifacts in this step of the analysis process were the PowerPoint file of family biographies and the digital photographs.

The PowerPoint file was the most interesting of these shared artifacts (see Fig. 4). It became the focus for revising and reiterating the deep understanding of the data that the team was building up about the domain through their shared conversations. It was the main working document for recording the emerging understandings of the grandparent/grandchild relationship, but due to its affordance for sharing with the whole group by projecting it on to a wall in the meeting room it became the central information receptacle around which conversations about meanings, motivations and interpretations were woven. The flexible and dynamic nature of the document afforded continual modification and reassessment of information, allowing the group to cooperatively rework the document and add revised understandings to it. The content of this artifact was the ethnographic record. The ethnographers used it to document their rich and concrete findings. The significance of this record for the software engineers was in the actual terminology used by the ethnographers. The engineers used these words to draw out representative notions and to abstract and generalize the findings. By its disposition in the meeting room, as a large, visible and clearly shared artifact, it triggered conversations between both communities of practice, communicating and consolidating their thoughts by bouncing ideas off each other, ideas that were enriched by the differing viewpoints present at the meetings.

The photographs acted as a shared artifact only in so far as they were used as a prop to help the ethnographers to communicate and discuss what was interesting about what was happening in the Magic Box exchanges (see Fig. 5). Their disposition in the meeting environment was the same as the PowerPoint file. They were presented electronically

Fig. 5. Photograph of Magic Box exchange projected on wall in.

to the group and projected on the wall, so in this way they afforded sharing by the whole group concurrently. Even though this was the only glimpse of Magic Box exchanged items that the team saw, they did not extract much understanding from the content of the photographs alone. Rather it was the stories that were told around these photographs and the interpretations brought to the table by the research pair that was of most value to the team. The significance of the artifact in terms of understanding the interactions that were taking place was not always obvious just from viewing it. Many required the context to be provided by the research pair. It was their close contact with a specific family that helped the researchers to recognize the significance of certain items. The researchers were able to present the photographs to the team, woven into the context, culture and past experiences of the family that the photograph alone could not convey. The stories were about the interesting and quirky things noticed by the researchers. They were about objects that generated tension, as well as happiness. Take for example the cow dung that was exchanged. Without the understanding of the joking nature of the father, and that it represented a day out at a farm, it could be perceived as an offensive offering.

During this whole process the software engineers were looking for higher-level categories to use in modeling the situation. They were looking for the kinds of roles that people were taking on in these stories that were being related to the team, and what was motivating their exchanges, for the purpose of capturing these in the models. These stories also provided persistent and memorable, yet unwritten, examples that helped the software engineers to identify and represent as roles, goals and quality goals in their models. So it were the stories, which remain unrecorded and only existed at that point in time, which communicated information from the ethnographic understanding to help build the software engineering model. Of course the significance of these stories to the ethnographers was in part recorded in the shared ethnographic record as well.

The videotapes of interviews although shown to the group did not become a shared artifact. Even though they afforded the same visual sharing by the group, they did not afford the concurrent conversations so important to the group, because videos take time to view and require auditory attention. Any discussion has to occur in retrospect. The ethnographers used the videos as confirmation of data they had already added into the PowerPoint record using their notes from the interview, and their conversations around the whiteboard. It was from this predigested data in the PowerPoint file that the software engineers drew their inspirations, not the interviews themselves.

5.2. Making the models

It was important for the software engineers to identify abstractions that could be generalized from the concrete probe data collected. Initial abstractions were gleaned from conversations at the team meetings. The software engineers listened to the ways in which the ethnographers were describing the activities of intergenerational play and the social dynamics underpinning those activities. Abstracting from the content in the situated context to the participants' intentions was important in identifying goals. Understanding the analysis of the participants' motivations for their activities came out of the team conversations. The software engineers then had to think about how to map motivations and intentions into goals or quality goals. The top-level Magic Box Goal Model, illustrated in Fig. 6, was the first model to be developed. This gives an overview of the goals, quality goals, and some of the roles initially identified.

In general, the study participants' motivations and intentions mapped to goals, and the abstraction and generalization of their intentions led to the discovery of roles. The highest-level goal in Fig. 4, Magic Box Goal Model, has five sub-goals: *Play*, *Gift*, *Show and Tell*, *Look*

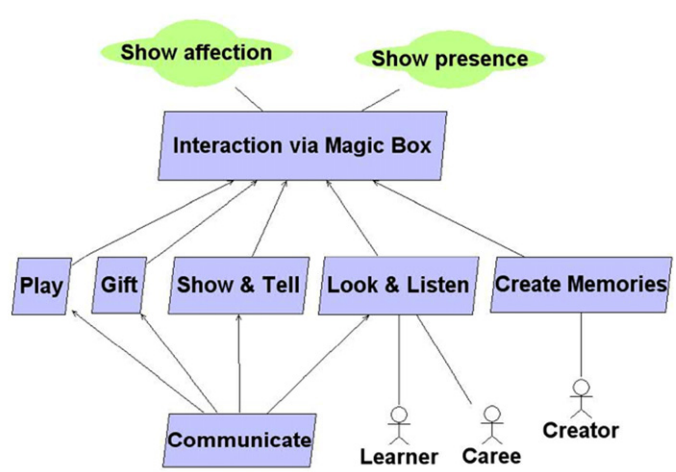

Fig. 6. Top-level Magic Box Goal Model.

and Listen and *Create Memories*. In this case study, play was identified as an important motivation for the Magic Box exchange. Storytelling was apparent in the data, as was gifting. Showing presence at a distance was also important. Play, gifting and storytelling are relatively concrete concepts and were identified as goals. Showing presence was an implicit motivation, achieved by doing something else in order to show presence. Therefore *showing presence* became a quality goal, as did *showing affection*.

During this phase the scrapbooks produced by the families and collected at the end of each family study were taken away from the group and analyzed in detail by a software engineer, to identify roles, goals and quality goals in different scenarios that could be pieced together from the data. The quality goals could be derived from analyzing the qualities implicit in episodes in the scrapbooks that could be identified as playing, gifting, showing and telling, and creating memories. In this way the scrapbooks became artifacts of the process, but not used cooperatively in team meetings. They were however used by different individuals at different times. This is probably due to the fact that scrapbooks afford viewing by a single person at a time. They are an incredibly rich and yet messy artifact, and really do return fragmentary clues about the lives and thoughts of the participants. They have pages, which affords some sense of temporality, in that the people creating them generally move to the next available page, but are strongly linked to the situation in which entries are made. They held interesting content, but the significance of that content was only partially realized when placed in the overall story of the family, woven over weeks of analysis. Some content was so personal it was never decoded. However, the software engineer's ability to understand and interpret the rich, fragmentary and heterogeneous nature of the data contained in the scrapbooks was a direct consequence of their participation in earlier team discussions. In this way, the scrapbook itself was not a shared object, but was interpretable as a result of the conversations motivated by other shared objects from the process.

5.3. Understanding the models

Once developed, the top-level goal model was then presented back to the team. The ROADMAP representation was introduced to the ethnographers, who were able to interpret the model. Opinions were then elicited from the team about the validity of the model. This involved informal conversations focused on the elements of the goal model. These conversations were recorded, both on tape and by note taking. The software engineers asked questions and clarified responses, as it was their responsibility to modify the model and present the next iteration to the team. At the same time they had to accept that the ethnographers perceived the presented analysis as deficient because of the loss of detail and richness from the probe data that the abstraction and generalization to create the model had necessitated. The purpose of this session was clearly understood by both parties as a presentation to discover what had been overlooked or wrongly captured and for the software engineers to gain a richer understanding of the nuances in interactions regarded as important by the ethnographers.

The session provided extremely rich feedback to the software engineers. Roles were one focus of the conversations. One role presented was *Family Member*, similar to a generic system user. This was shown to be completely inadequate. Firstly, each family member interacts with other family members differently. Interaction between grandparents and grandchildren tends to be asymmetrical, with grandparents taking the initiative and being more reflective. Also, parents need to be represented, since at least the younger children need support. The interim idea of having a role for each family member was floated: *Grandparent*, *Child* and *Parent*. Further discussion moved beyond this concept. People choose to play different roles at different times. Some grandparents like to get down to the child's level on occasion. Alternatively, for example with storytelling, the child must only say "It's my turn now", for the roles to be switched. The refined requirements analysis gained a series of complementary roles such as *Carer/Caree*, *Teacher/Learner*, as well as the role *Coach*, which could be a parent. This cooperative analysis and review of the goal model diagram by the team resulted in the identification of the roles shown in Fig. 6: *Learner*, *Caree*, and *Creator*.

In this step of the analysis, the ROADMAP goal model became a shared artifact. It was presented to the group electronically and projected on the wall in the same way as the PowerPoint file and the photographs. This afforded viewing by the team as a whole. The software engineers hoped that the simplicity of the notation would facilitate immediate understanding by the ethnographers, but it became evident to the software engineer who presented it, that for the ethnographers to gain meaning from the diagram it was necessary to remind them of the detailed stories on which the model was based. The stories that formed that basis of conversations during the first step of the process became the examples given by the software engineer to explain the abstracted goal model. For example, photographs of biscuits, cow dung, a toy pony and vegetables where generalized in the Magic Box goal model to the term *gift*, but each gift had a much more interesting story about why it was chosen, who it was for and the motivation for sending it. These stories had to be verbally attached to the model. The quality goals associated with the *Gift* goal were found to be not detailed enough, in their simple label, to carry an understanding of the social motivations of the "gifter" through to design.

Some of the ethnographers viewed the ROADMAP goal model simply as a transmission object, and did not feel that it acted as a shared artifact. Unlike the PowerPoint file, cooperative refinement of the model could not be made because it was created in a different software package and imported as a single image. It may have been this lack of visible collaborative analysis during the meeting that made the ROADMAP goal model artifact feel as if it belonged to the software engineers who created it, and who had full control over modifying it. It was not just the content of the model, but also the significance that each community of practice could attach to it that was a problem. Some ethnographers felt that the model did not provide an interpretably flexible object for facilitating articulation of their own understandings. It was too simplistic and too structured. At the same time, others claimed that the abstracted terms that came from the ROADMAP model became items for debate in the team meeting, and in this way triggered a reevaluation of their own understanding of the situation being modeled. For example, when the quality goal *showing presence* was reflected back to the team, many discussions ensued which questioned exactly what it meant to these families to be showing presence to each other, resulting in refined interpretations of the objects that had been exchanged by Magic Box, and modification to the ethnographic record held in the PowerPoint file. In this way there was reciprocity in the analytical process and viewpoints were challenged by the complementary understandings being shared through the artifact. The fact that it was only displayed in the meeting environment, and created and modified elsewhere, limited the groups feeling of ownership over the artifact.

5.4. Refining the models

Once abstractions were identified and discussed, the software engineers returned to the richness of the cultural probe data in order to flesh out more detailed lower level requirements. They examined the cultural probe returns, going through the raw data to find concrete examples of potential abstractions to confirm their validity. Several scrapbooks were re-examined. One story of grandparents' migration to Australia was used to flesh out the *Show and Tell* Goal Model diagram shown in Fig. 7. Reviewing the scrapbooks provided evidence of interaction patterns, reflecting the roles such as *Teacher* described above. One child pasted letters from his grandmother into his scrapbook. Every letter was educative, explaining to the child how to perform particular tasks.

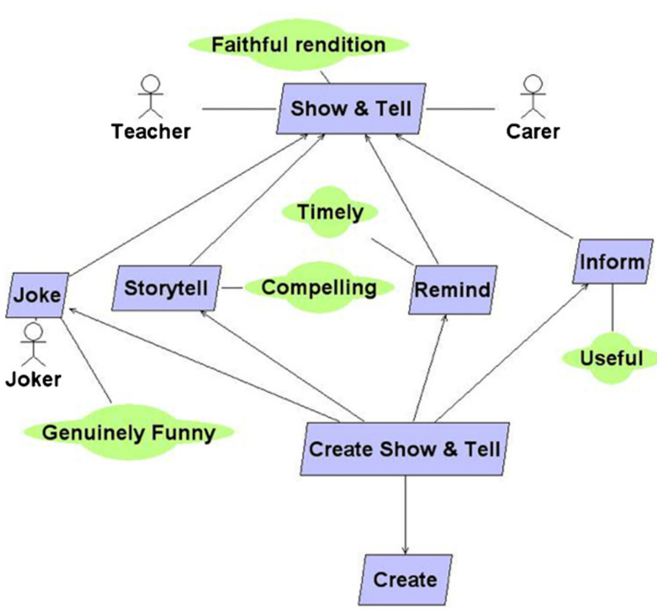

Fig. 7. Show and Tell Goal Model for the Magic Box.

This more detailed goal model discussion resulted in the identification of the additional role of *Joker*. It also resulted in the identification of five additional quality goals: *faithful rendition*, *timely*, *compelling*, *genuinely funny* and *useful*. As shared artifacts, these lower level models played the same role in the meetings as the high-level goal models. When presented back to the multidisciplinary team they received similar responses as in the previous step, with the ethnographers happy to collaborate in the discovery of appropriate quality goal labels, but continuing to feel that the quality goals did not capture or communicate the depth of understanding of the social interactions that they felt was imperative. Although the software engineers felt that the ROADMAP representation was more capable of representing social aspects than any other requirements elicitation methodology they had worked with.

5.5. Adding scenarios to the models

The cultural probe data and the feedback from the team discussions prompted the refinement of the AOSE modeling process by adding scenarios to the ROADMAP methodology. Scenarios were not originally part of ROADMAP and were added in direct response to collaborative conversations advocating the need for richer design data to be captured with the AOSE models. Design scenarios were already part of the Prometheus methodology, for example the Storytell scenario shown in Fig. 8. In the case study, creating design scenarios clarified that in the process storytelling, if the goal of a family member is to hear a story, then it requires the percept that another family member is willing to tell a story before a story can be told and listened to. To create this design artifact, the informal scenarios presented along with the ROADMAP models to the team were used. This led to the formalization of requirements scenarios into the method to support the goal model diagrams, by modularizing more detailed understandings of social interactions.

In a way, the scenarios that were delivered as explanations by the software engineer when presenting the model could be considered as shared artifacts, albeit ephemeral ones, that provided more content and significance to the ethnographers than the models did. The practice of abstracting complex concepts into simple terms was necessary for the software engineers to construct generalized models, but this caused the ethnographers to feel that, although the model represented the right kind of idea, the richness of the interaction had been lost.

Scenario Storytell scenario

Name	Storytell scenario
Description	
Priority	Not Specified
Stakeholders	
Initiated by	System
Trigger	
Steps	(see below)
Variation	

#	Type	Name	Role	Description	Data used	Data produced
1	Percept	Desire to Play	Family Member	Family member decides they want to hear a story		
2	Goal	Play	Family Member	Play in the form of hearing a story		
3	Action	Interaction invitation	Interaction Mediator			
4	Action	Send Item	Sender	send a request to hear a story from another family member		
5	Percept	Item arrives	Receiver	Other family member replies that will tell story		
6	Goal	Tell	Storyteller	Story is told		
7	Goal	Hear	Listener	Family member listens to story		

Fig. 8. Storytelling scenario for Magic Box case study as recorded.

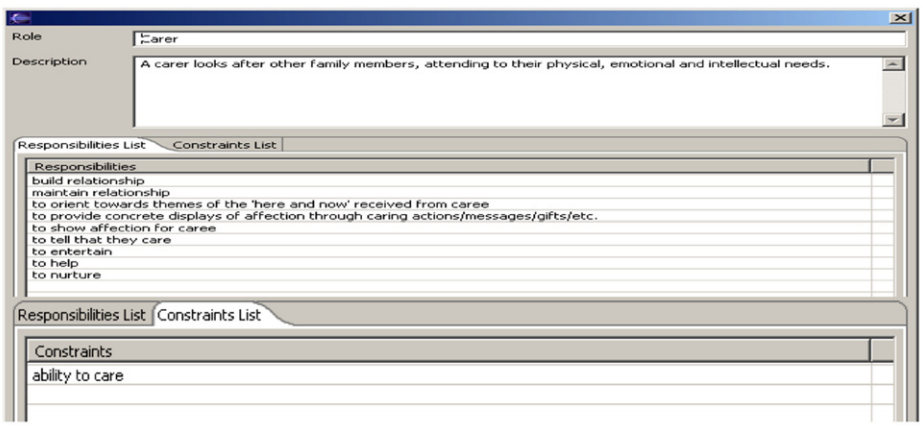

Fig. 9. Role model for the carer role.

The ethnographers did not regard the model as 'incorrect'—they just felt that it did not seem to capture the subtleties and richness of the observed family interactions. Revisiting the example of the goal *Gift*, the gifts exchanged were more than just objects, they held qualities of time investment, labour, effort, meaning and thought from the giver. The ethnographers felt that these qualities of the gift were lost in the model. To some extent the addition of scenarios to the models, which explicitly documented the stories that currently only existed verbally in the meeting, was proposed as a way to alleviate that concern. Scenarios have been used to provide a shared vocabulary among people involved in systems development and to translate understanding of a rich variety of roles in the system from different viewpoints, and to diverse communities (Go and Carroll, 2004). The requirements scenarios were related to goals and thereby could carry details through to the design phase to enrich the software designers understanding of the goals and quality goals while retaining the necessary abstraction at goal model level.

5.6. Documenting quality goals

After the refinement of the scenarios, the software engineers and the ethnographers no longer met as a team. The software engineers had gained all they needed from the cultural probe data, from team conversations, and from direct feedback on the models. The main purpose of these, or similar, models in AOSE methodologies is to document the aspects of a system that are required toward designing and implementing a software appliance, prototype or technology probe. The ROADMAP requirements analyses and models were discussed informally with other AOSE researchers to confirm their fidelity as models to be passed to the design phase. To ensure that the quality goals remained part of the analysis, role models were developed where the quality goals formed an important part of the constraints imposed on the roles in a system, for example, the quality goal of *caring*, meaning the person has a caring nature, formed a constraint in the role of the *Carer* from the *Show and Tell* Goal Model, as shown in Fig. 9.

Constraints listed in the role model are used to document how a quality goal is to be achieved. In this case the role *Carer* has the constraint "ability to care", that represents important detail about the quality goal *caring*. This is the mechanism that passes the quality goal requirements of the system to the software design phase.

6. Discussion

After completing the analysis of the cultural probe data and requirements elicitation we had an organized ethnographic record of the situation of intergenerational play, and a set of ROADMAP diagrams of sufficient fidelity for designing a Magic Box socio-technical system.

In this discussion, we reflect on our process and examine how the shared artifacts were used by the multidisciplinary team to 'bridge the gap' between ethnography and software engineering. We first explore the roles that shared artifacts played in the process of analysis and requirements elicitation and then examine the contribution of the quality goals in providing a transformation mechanism for the ethnographic findings of our case study into the realm of software engineering.

6.1. Shared artifacts

The key artifacts in the process were the ethnographic PowerPoint record, and the AOSE ROADMAP goal diagrams. These shared artifacts presented different views of the same situation but both represented a reduction in complexity of the raw data. They both facilitated communication between the two communities of practice. Both evolved as a participatory activity, with team conversations facilitating the cooperative identification of aspects of both artifacts. In moving from probe data to data models, the artifacts helped both communities to reflect on their own practice and thinking, while at the same time, sustaining different interpretations that enabled conversations to flow. The shared artifacts were complex enough to enable lines of work to cohere, while usefully maintaining multiple interpretations for the different communities involved in the design process.

These two artifacts, displayed on a wall with a data projector, were clearly visible to all participants in the meeting and thereby provoked conversations that the whole team participated in. The clarification of terminology across disciplines and the exchange of differing interpretations of the data that happened in meetings meant that artifacts that would otherwise have been seen as clearly belonging to one or other community became shared, in the sense that they were consumable by both. Collaborative modification of an artifact gave team members more of a sense of ownership over that artifact. When a term was discussed in a meeting and then appeared both in the ethnographic record (PowerPoint file) and as a label in the goal model (ROADMAP) diagram reciprocity was established and transformation achieved.

The PowerPoint file was the focus of meetings where the ethnographers presented their experience with the families from the previous week, and discussed the probe returns in detail. This artifact helped communicate stories of families while maintain the complex, organic and slightly messy qualities of the probe data for both communities of practice. The pair of researchers that worked closely with a particular family would present the photographs of the Magic Box exchanges back to the group and their interpretations would form an entry in the ethnographic record. This would be discussed and modified in respect to the overall understanding of the situation developing in the team over time and the personal and professional experiences of both the ethnographers and software engineers present at the meeting. They would review interview notes on a whiteboard and watch video tapes, but the collaborative analysis would happen as this data was organized and added into the continually updated PowerPoint file. The affordance of this artifact to be immediately updated by the team gave this shared artifact meaning for both the ethnographers and the software engineers, as both contributed to, and experienced its evolution.

The conversations that happened in meetings with the ROADMAP models as shared artifacts were enriched by the presence of the models. The actual modeling provided clarification to the conversations, and the push and shove in these conversations extended the scope of the understanding derived from the data. For example, in the original Magic Box version, the quality goal *accurate* constrained the goal *Remember*. In the exact world of software implementation, the meaning of "accurate" is considered obvious. However, questions immediately arose during presentation of the Magic Box requirements analysis to the ethnographers: What exactly should be remembered? What does the quality goal *accurate* refer to, accuracy in sending to the correct recipient, accuracy of content, accuracy in timing? Perhaps a more appropriate term would be *faithful rendition* as the memory of a story, where the intentions of the storyteller are faithfully captured?

At the same time, there was a feeling amongst the ethnographers that the ROADMAP models in their current form, even with the constructs of scenarios and quality goals, were still not capable of delivering detail about the social interactions to the design process, necessary for engineering playful and interesting socio-technical systems. The flexibility and high-level nature of ROADMAP enabled software engineers to present very high-level abstractions in respect to the goals and roles of the system. The simplicity of the diagrams presented to the multidisciplinary team was critical to facilitating discussion of the concepts portrayed. It was agreed, however, that agent concepts provided a distinct advantage over object-oriented concepts when dealing with socially oriented requirements engineering. The concept of agents was shown to facilitate understanding of software design by members of the multidisciplinary team who were initially completely disinterested in the software engineering process.

6.2. Quality goals

In the agent-oriented approach, quality goals were able to capture the essence of the kinds of non-functional 'social requirements' that were identified in the cultural probe data, and to make them available to the system design process. Ambiguity and uncertainty in the ethnographic findings were well served by the quality goals in ROAD-MAP. In reflecting on the process it can be seen that quality goals that have paradoxes hidden in them, created new questions. These questions stretched the conversations on both sides. The ethnographers learned things through these conversations, and in the development of the models that they felt they would not have gained otherwise. For example, a similar discussion was provoked by the proposed quality goals for storytelling. The sub-goal to *Tell* a story was initially qualified by *interesting* and the sub-goal to *Hear* a story was qualified by *understand*. Again the questions were immediate: What does "understand" mean? Each listener understands something different from a story. What does "interesting" mean? Why should a story be interesting for the teller as opposed to the listener? What about stories, which are told and re-told countless times? The outcome of this discussion was that a story should be *compelling*, meaning that a storyteller wants to keep on telling and a listener wants to keep on listening. Both are enjoying the playful interaction.

In the end, quality goals were confirmed to be useful for capturing intangible requirements and goals, as often encountered in social contexts. Through the use of goals, roles and specifically quality goals, we feel that AOSE requirements engineering was performed using the results of cultural probe ethnographic work without losing the elements of subjectivity, empathetic interpretation and uncertainty that are central to the cultural probe approach.

7. Conclusion

The current shift in focus toward observing the home environment rather than the workplace in order to understand interactions around and with technology in the home has brought a shift in the methods required for observation in this different situational context. The use of cultural probes, with their emphasis on fragmentary, subjective, interpretive data to inspire speculative design for new uses of technology, is a valuable observational method for studying social interactions.

On the basis of a case study in cultural probe work concerning intergenerational play, we discovered the importance of shared artifacts in moving forward from the results of cultural probe studies towards a requirements analysis, while retaining central aspects of cultural probe research, such as subjectivity, uncertainty and inspirational design.

The process outlined in this paper, of eliciting socially oriented requirements for software engineering purposes, demonstrates the value of an ongoing relationship between ethnographers and software engineers during data analysis and requirements elicitation. Our process was instituted around conversations stimulated by artifacts shared at multidisciplinary meetings. The artifacts in these meetings, (i.e., returned probe items, scrapbooks, videos of interviews, photographs, family biographies and the ROAD-MAP models), played a central role in the analysis phase and mediated a reciprocal relationship between the place and activity concerns of HCI researchers, and the SE focus on domain specific and application specific models. They motivated conversations that shaped the analysis process, enabling the cooperative identification of quality goals and producing a robust and serious account of social interactions in the domestic setting toward the design of socially oriented technology.

In the end, artifacts that helped bridged between the practices of ethnographers and the representations of the software engineers where those that acted as a shared resource because they could be created, modified and refined cooperatively during meetings. All who participated in the process agreed that the greater diversity of viewpoints applied to the data, the better, demonstrating the success of the multidisciplinary approach. The shared and diverse experiences of all members of the team contributed to engagement in the conversations, and provided a deeper relationship to the data for both the ethnographers and the software engineers. Through this familiarity with the raw data, the software engineers could access and represent in quality goals ambiguous aspects of the social interactions detailed in the ethnographic account, thereby engineering the social.

Acknowledgments

Related work was supported by SITCRC, the Smart Internet Technology Cooperative Research Center funded by the Australian Government and industry partners. The project built on work from the SITCRC project 'Intergenerational Distributed Family'. We would like to thank the following researchers for their contributions in sharing their experiences in that project: Martin Gibbs, Hilary Davis, Peter Francis, and Peter Benda.

The current work is funded by the Australian Research Council discovery grant DP0880810 'Socially Oriented Requirements Engineering-Software Engineering meets Ethnography', with Chief Investigators: Prof. Leon Sterling, Dr. Frank Vetere, A/Prof. Steve Howard.

References

Arnold, M., 2004. The connected home: probing the effects and affects of domesticated ICTs. In: Proceedings of PDC2004, Toronto, ACM.

Beck, K., 2000. Extreme Programming Explained: Embrace Change. Addison-Wesley, Reading, MA.

Bell, G., Blythe, M., Sengers, P., 2005. Making by making strange: defamiliarization and the design of domestic technologies. ACM Transactions on Computer–Human Interaction 12 (2), 149–173.

Bell, G., Dourish, P., 2007. Back to the shed: gendered visions of technology and domesticity. Personal and Ubiquitous Computing 11 (5), 373–381.

Benford, S., Schnadelbach, H., Koleva, B., Anastasi, R., Greenhalgh, C., Rodden, T., Green, J., Ghali, A., Pridmore, T., Gaver, B., Boucher, A., Walker, B., Pennington, S., Schmidt, A., Gellersen, H., Steed, A., 2005. Expected, sensed, and desired: a framework for designing sensing-based interaction. ACM Transactions on Computer–Human Interaction 12 (1), 3–30.

Beyer, H., Holtzblatt, K., 1998. Contextual Design: Defining Customer-Centred Systems. Morgan Kaufmann, San Francisco.

Blythe, M., Monk, A., Park, J., 2002. Technology biographies: field study techniques for home use product development. In: Proceedings of CHI '02. Minneapolis, MN, USA, pp. 658–659.

Boehner, K., Vertesi, J., Senger, P., Dourish, P., 2007. How HCI interprets the probes. In: Proceedings of CHI '07. ACM, pp. 1077–1086.

Button, G., Dourish, P., 1996. Technomethodology: paradoxes and possibilities. In: Proceedings of CHI 96. ACM, Vancouver, Canada, pp. 19–26.

Castro, J., Kolp, M., Mylopoulos, J., 2001. A requirements-driven development methodology. In: Proceedings of CaiSE 2001. Springer, Interlaken, Switzerland, pp. 108–123.

Cheverst, K., Clarke, K., Dewsbury, G., Hemmings, T., Hughes, J., Rouncefield, M., 2003. Design with care: technology, disability and the home. In: Harper, R. (Ed.), Inside the Smart Home. Springer, London, pp. 163–180.

Christensen, C., 1997. The Innovator's Dilemma: When New Technologies Cause Great Firms to Fail. Harvard Business School Press, Boston.

Chung, L.K., Nixon, B.A., Yu, E., Mylopoulos, J., 2000. Non-Functional Requirements in Software Engineering. Kluwer Publishing.

Constantine, L., Biddle, R., Noble, J., 2003. Usage-centered design and software engineering: models for integration. In: Proceedings of ICSE 2003, IFIP, Portland, USA, pp. 106–113.

Constantine, L., Lockwood, L., 2003. Usage-centered software engineering: an agile approach to integrating users, user interfaces, and usability into software engineering practice. In: Proceedings of ICSE03. Portland, OR, USA, pp. 746–747.

Crabtree, A., 2000. Talking work: language-games, organisations and computer supported cooperative work. Computer Supported Cooperative Work 9 (2), 215–237.

Crabtree, A., 2004. Design in the absence of practice: breaching experiments. In: Proceedings of DIS 2004. Cambridge, MA, USA, pp. 59–68.

Crabtree, A., Hemmings, T., Rodden, T., 2002. Pattern-based support for interactive design in domestic settings. In: Proceedings of DIS 2002. London, England, pp. 265–276.

Crabtree, A., Nichols, D., O'Brien, J., Rouncefield, M., Twidale, M., 2000. Ethnomethodologically informed ethnography and information system design. Journal of the American Society for Information Science 51 (7), 666–682.

Crabtree, A., Rodden, T., Benford, S., 2005. Moving with the times: IT research and the boundaries of CSCW. Computer Supported Cooperative Work 14 (3), 217–251.

Cunningham, S.J., Jones, M., 2005. Autoethnography: a tool for practice and education. In: Proceedings of CHINZ '05. Auckland, New Zealand, pp. 1–8.

Davila, T., Epstein, M., Shelton, R., 2006. Making Innovation Work: How to Manage It, Measure It, and Profit from It. Wharton School Publishing, Upper Saddle River.

de Paula, M., Barbosa, S., de Lucena, C., 2005. Conveying human–computer interaction concerns to software engineers through an interaction model. In: Proceedings of CLIHC '05. Cuernavaca, Mexico, pp. 109–119.

Diggins, T., Tolmie, P., 2003. The 'adequate' design of ethnographic outputs for practice: some explorations of the characteristics of design resources. Personal and Ubiquitous Computing 7, 147–158.

Frohlich, D., Kraut, R., 2003. The social context of home computing. In: Harper, R. (Ed.), Inside the Smart Home. Springer, London, pp. 127–162.

Gaver, B., 2001. Designing for ludic aspects of everyday life, ERCIM News. No. 47, October 2001. 〈http://www.ercim.org/publication/Ercim_News/enw47/gaver.html〉 (last accessed 1/6/2008).

Gaver, W., Boucher, A., Pennington, S., Walker, B., 2004. Cultural probes and the value of uncertainty. Interactions 11 (5), 53–56.

Gaver, B., Dunne, T., Pacenti, E., 1999. Design: cultural probes. Interactions 6 (1), 21–29.

Go, K., Carroll, J., 2004. The blind men and the elephant: views of scenario-based system design. Interactions 11 (6), 44–53.

Goguen, J., 1993. Social issues in requirements engineering. In: Proceedings of RE '93. IEEE Computer Society, Bonn, Germany, pp. 194–195.

Goguen, J., Linde, C., 1993. Techniques for requirements elicitation. In: Proceedings of RE '93. IEEE Computer Society, Bonn, Germany, pp. 152–164.

Graham, C., Rouncefield, M., Gibbs, M., Vetere, F., Cheverst, K., 2007. How probes work. In: Proceedings of OZCHI 2007. ACM, Adelaide, Australia, pp. 29–37.

Haines, V., Mitchell, V., Cooper, C., Maguire, M., 2007. Probing user values in the home environment within a technology driven Smart Home project. Personal and Ubiquitous Computing 11, 349–359.

Halverson, C., Erickson, T., Ackeman, M., 2004. Behind the help desk: evolution of a knowledge management system in a large organization. In: Proceedings of CSCW'04. Chicago, IL, USA, pp. 304–314.

Harper, R., 2003. Inside the Smart Home: ideas, possibilities and methods. In: Harper, R. (Ed.), Inside the Smart Home. Springer, London, pp. 1–14.

Harper, R., Rodden, T., Rogers, Y., Sellen, A., 2008. Being Human: Human–Computer Interaction In The Year 2020. Microsoft Research Ltd.

Hemmings, T., Clarke, K., Crabtree, A., Rodden, T., Rouncefield, M., 2002. Probing the probes. In: Proceedings of PDC 2002. Computer Professionals Social Responsibility, Malmo, Sweden, pp. 42–50.

Hindus, D., 1999. The importance of homes in technology research. In: Proceedings of CoBuild'99. Springer, Pittsburgh, USA, pp. 199–207.

Howard, S., Kjeldskov, J., Skov, M., 2007. Pervasive computing in the domestic space. Personal and Ubiquitous Computing 11, 329–333.

Hughes, J., King, V., Rodden, T., Andersen, H., 1994. Moving out from the control room: ethnography in system design. In: Proceedings of CSCW '94. ACM, Chapel Hill, USA, pp. 429–439.

Hughes, J., O 'Brien, J., Rodden, T., Rouncefield, M., Blythin, S., 1997. Designing with ethnography: a presentation framework for design. In: Proceedings of DIS '97. ACM, Amsterdam, The Netherlands, pp. 147–158.

Hughes, J., King, V., Rodden, T., Andersen, H., 1995a. The role of ethnography in interactive systems design. Interactions 2 (2), 56–65.

Hughes, J., O'Brien, J., Rodden, T., Rouncefield, M., Sommerville, I., 1995b. Presenting ethnography in the requirements process. In: Proceedings of RE '95. IEEE, York, England, pp. 27–34.

Hutchinson, H., Mackay, W., Westerlund, B., Bederson, B., Druin, A., Plaisant, C., Beaudouin-Lafon, M., Conversy, S., Evans, H., Hansen, H., Roussel, N., Eiderback, B., Lindquist, S., Sundblad, Y., 2003. Technology probes: inspiring design for and with families. In: Proceedings of CHI 2003. ACM, Fort Lauderdale, FL, pp. 17–24.

Iqbal, R., Gatward, R., James, A., 2005a. A general approach to ethnographic analysis for systems design. In: Proceeding of SIGDOC 2005. ACM, Coventry, UK, pp. 34–40.

Iqbal, R., James, A., Gatward, R., 2005b. Designing with ethnography: an integrative approach to CSCW design. Advanced Engineering Informatics 19, 81–92.

Juan, T., Pearce, A., Sterling, L., 2002. ROADMAP: extending the Gaia methodology for complex open systems. In: Proceedings of AAMAS 2002. ACM, Bologna, Italy, pp. 3–10.

Kjeldskov, J., Gibbs, M., Vetere, F., Howard, S., Pedell, S., Mecoles, K., Bunyan, M., 2005. Using cultural probes to explore mediated intimacy. Australian Journal of Information Systems 1 (12), 102–115.

Kjeldskov, J., Nielsen, C.M., Overgaard, M., Pedersen, M.B., Stage, J., Stenild, S., 2006. Designing a mobile communicator: combining ethnography and object-oriented design. In: Proceedings of OZCHI 2006. ACM, Sydney, Australia, pp. 95–102.

Kuan, P., Karunasekera, S., Sterling, L., 2005. Improving goal and role oriented analysis for agent based systems. In: Proceedings of ASWEC 2005. IEEE, Brisbane, Australia, pp. 40–47.

Larman, C., Basili, V., 2003. Iterative and incremental development: a brief history. Computer 36 (6), 47–56.

Lee, C., 2007. Boundary negotiating artifacts: unbinding the routine of boundary objects and embracing chaos in collaborative work. Computer Supported Cooperative Work 16 (3), 307–339.

Martin, R.C., 2003. Agile Software Development: Principles, Patterns, and Practices. Prentice-Hall, NJ, USA.

Martin, D., Sommerville, I., 2004. Patterns of cooperative interaction: linking ethnomethodology and design. ACM Transactions on Computer–Human Interaction 11 (1), 59–89.

Millen, D.R., 2000. Rapid ethnography: time deepening strategies for HCI field research. In: Proceedings of DIS '00. ACM, Brooklyn, New York, pp. 280–286.

Neale, D., Carroll, J., Rosson, M., 2004. Evaluating computer-supported cooperative work: models and frameworks. In: Proceedings of CSCW '94. ACM, Chapel Hill, USA, pp. 112–121.

Neto, G., Gomes, A., Castro, J., Sampaio, S., 2005. Integrating activity theory and organizational modeling for context of use analysis. In: Proceedings of LIHC'05. ACM, Cuernavaca, Mexico, pp. 301–306.

Norman, D., 1999. The Invisible Computer: Why Good Products Can Fail, the Personal Computer Is So Complex, and Information Appliances Are the Solution. MIT Press, Cambridge, MA, USA.

Nuseibeh, B., Easterbrook, S., 2000. Requirements engineering: a ROADMAP. In: Finkelstein, A. (Ed.), The Future of Software Engineering. ICSE, pp. 35–46.

O'Brien, J., Rodden, T., 1997. Interactive systems in domestic environments. In: Proceedings of DIS '97. ACM, Amsterdam, The Netherlands, pp. 247–259.

Ormerod, T., Morley, N., Ball, L., Langley, C., Spenser, C., 2003. Using ethnography to design a mass detection tool (MDT) for the early discovery of insurance fraud. In: Proceedings of CHI 2003. ACM, Fort Lauderdale, FL, pp. 650–651.

Paay, J., 2008. From ethnography to interface design. In: Lumsden, J. (Ed.), Handbook of Research on User Interface Design and Evaluation for Mobile Technology. Idea Group Inc (IGI), PA, USA, pp. 1–15.

Padgham, L., Winikoff, M., 2004. Developing Intelligent Agent Systems: A Practical Guide. Wiley, New York, ISBN 0-470-86120-7.

Rahwan, I., Juan, T., Sterling, L., 2006. Integrating social modelling and agent interaction through goal-oriented analysis. Computer Systems Science & Engineering 3, 87–98.

Raijmakers, B., Gaver, W., Bishay, J., 2006. Design documentaries: inspiring design research through documentary film. In: Proceedings of DIS 2006. ACM, University Park, Pennsylvania, USA, pp. 229–238.

Ramduny-Ellis, D., Dix, A., Rayson, P., Onditi, V., Sommerville, I., Ransom, J., 2005. Artefacts as designed, artefacts as used: resources for uncovering activity dynamics. Cognition, Technology and Work 7, 76–87.

Rouncefield, M., Crabtree, A., Hemmings, T., Rodden, T., Cheverst, K., Clarke, K., Dewsbury, G., Hughes, J., 2003. Adapting cultural probes to inform design in sensitive settings. In: Proceedings of OzCHI 2003. Ergonomics Society of Australia, Brisbane, Australia, pp. 4–13.

Schraefel, M., Hughes, G., Mills, H., Smith, G., Frey, J., 2004. Making tea: iterative design through analogy. In: Proceedings of DIS '04, Cambridge, MA, USA, pp. 49–58.

Sengers, P., Gaver, W., 2006. Staying open to interpretation: engaging multiple meanings in design and evaluation. In: Proceedings of DIS '06, University Park, PA, USA, pp. 99–108.

Sharp, H., Rogers, Y., Preece, J., 2007. Interaction Design: Beyond Human–Computer Interaction. Wiley, West Sussex, England.

Simonsen, J., 1997. Using ethnography in contextual design. Communications of the ACM 40 (7), 82–88.

Simonsen, J., Kensing, F., 1998. Make room for ethnography in design!. Journal of Computer Documentation 22 (1), 20–30.

Sommerville, I., 2007. Software Engineering: (Update), eighth ed. Pearson Education Limited.

Sousa, K., Furtado, E., 2003. RUPi—a unified process that integrates human-computer interaction and software engineering. In: Proceedings of ICSE 2003. Workshop on Bridging the Gap between SE and HCI. IFIP, Portland, USA, pp. 41–48.

Star, S.L., Griesemer, J., 1989. Institutional ecology, 'translations' and boundary objects: amateurs and professionals in Berkeley's museum of vertebrate zoology, 1907–39. Social Studies of Science 19 (3), 387–420.

Suchman, L., 1987. Plans and Situated Actions: The Problem of Human–Machine Communication. Cambridge University Press, Cambridge, NY.

Surendra, N., 2008. Using an ethnographic process to conduct requirements analysis for agile systems development. Information and Technology Management 9, 55–69.

Sutcliffe, I., 2005. Applying small group theory to analysis and design of CSCW systems. In: Proceedings of HSSE 2006. ACM, St. Louis, MI, USA, pp. 1–6.

Taylor, A., Swan, L., 2005. Artful systems in the home. In: Proceedings of CHI 2005. ACM, Portland, OR, USA, pp. 641–650.

Terrenghi, L., Hilliges, O., Butz, A., 2007. Kitchen stories: sharing recipes with the Living Cookbook. Personal and Ubiquitous Computing 11, 409–414.

Thangarajah, J., Padgham, L., Winikoff, M., 2005. Prometheus design tool. In: Proceedings of the AAMAS '05. The Netherlands, pp. 127–128.

Tolmie, P., Pycock, J., Diggins, T., MacLean, A., Karsenty, A., 2003. Towards the unremarkable computer: making technology at home in domestic routines. In: Harper, R. (Ed.), Inside the Smart Home. Springer, London, pp. 183–206.

van Lamsweerde, A., Letier, E., 2004. From object orientation to goal orientation: a paradigm shift for requirements engineering. In: Wirsing, M., Knapp, A., Balsamo, S. (Eds.), Radical Innovations of Software and Systems Engineering in the Future. Revised Papers from RISSEF 2002, LNCS 2941. Springer, pp. 325–340.

Venkatesh, A., 1996. Computers and other interactive technologies for the home. Communications of the ACM 39 (12), 47–54.

Vetere, F., Davis, H., Gibbs, M., Howard, S., 2009. The Magic Box and collage: responding to the challenge of distributed intergenerational play. International Journal of Human Computer Studies 67 (2), 165–178.

Vetere, F., Feltham, F., 2007. Bringing emotion and physicality to domestic ICTs: interview with Steven Kyffin. Personal and Ubiquitous Computing 11, 335–337.

Viller, S., Sommerville, I., 1999. Coherence: an approach to representing ethnographic analyses in systems design. Human–Computer Interaction 14, 9–41.

Viller, S., Sommerville, I., 2000. Ethnographically informed analysis for software engineers. International Journal of Human–Computer Studies 53, 169–196.

Vyas, D., Dix, A., 2007. Artefact ecologies: supporting embodied meeting practices with distance access. In: Proceedings of UbiComp 2007 Workshops. University of Innsbruck, Innsbruck, Austria, pp. 117–122.

Vyas, D., de Groot, S., van der Veer, G., 2006. Understanding the academic environments: developing personas from field-studies. In: Proceedings of ECCE 2006. ACM, Zurich, Switzerland, pp. 119–120.

Walenstein, A., 2003. Finding boundary objects in SE and HCI: an approach through engineering-oriented design theories. In: Proceedings of ICSE 2003. Workshop on Bridging the Gap between SE and HCI. IFIP, Portland, USA, pp. 92–99.

Wenger, E., 1998. Communities of Practice: Learning, Meaning and Identity. Cambridge University Press, Cambridge, UK.

Wiltshire, M., 2003. Where SE and HCI meet: a position paper. In: Proceedings of ICSE 2003, Workshop on Bridging the Gap between SE and HCI. IFIP, Portland, USA, pp. 57–60.

Wooldridge, M., 1999. Intelligent agents. In: Weiss, G. (Ed.), Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence. MIT Press, Cambridge, MA, pp. 27–78.

Wooldridge, M., Jennings, N., Kinny, D., 2000. The Gaia methodology for agent-oriented analysis and design. Journal of Autonomous Agents and Multi-Agent Systems 3 (3), 285–312.

Yu, E., 1995. Modelling strategic relationships for process reengineering. Ph.D. Thesis, Department of Computer Science, University of Toronto, Canada.

Zafiroglu, A., Chang, M., 2007. Digital homes on wheels: designing for the unimagined home. Personal and Ubiquitous Computing 11, 395–402.