Reframing the design of context-aware computing

Margot Brereton
Queensland University
of Technology
2 George St.
GPO Box 2434
Brisbane QLD 4001
Australia
m.brereton@qut.edu.au

Hadi Mirisaei,
Queensland University
of Technology
2 George St.
GPO Box 2434
Brisbane QLD 4001
Australia
s.mirisaei@qut.edu.au

Sunil Ghelawat
Queensland University
of Technology
2 George St.
GPO Box 2434
Brisbane QLD 4001
Australia
s.ghelawat@qut.edu.au

Paul Roe
Queensland University
of Technology
2 George St.
GPO Box 2434
Brisbane QLD 4001
Australia
p.roe@qut.edu.au

From location-aware computing to mining the social web, representations of context have promised to make better software applications. The opportunities and challenges of context-aware computing from representational, situated and interactional perspectives have been well documented, but arguments from the perspective of design are somewhat disparate. This paper draws on both theoretical perspectives and a design framing, using the problem of designing a social mobile agile ridesharing system, in order to reflect upon and call for broader design approaches for context-aware computing and human-computer Interaction research in general.

Context- aware computing, design, reframing, mobile social computing, human-computer interaction

1. INTRODUCTION

For more than a decade research in the field of context aware computing has aimed to find ways to exploit situational information that can be detected by mobile computing and sensor technologies. The goal is to provide people with new and improved applications, enhanced functionality and better use experience (Dey, 2001). Early applications focused on representing or computing on physical parameters, such as showing your location and the location of people or things around you. Such applications might show where the next bus is, which of your friends is in the vicinity and so on. With the advent of social networking software and microblogging sites such as Facebook and Twitter, recommender systems and so on context-aware computing is moving towards mining the social web in order to provide better representations and understanding of context, including social context.

In this paper we begin by recapping different theoretical framings of context. We then discuss the problem of context- aware computing from a design perspective.

2. CONTEXT- AWARE COMPUTING

Context-aware computing was first discussed by Schilit and Theimer (1994) as software that

“adapts according to its location of use, the collection of nearby people and objects, as well as changes to those objects over time”. Context aware computing arose, not only to exploit new possibilities but also to address some of the known challenges in human-machine interaction.

Suchman's 1987 classic observational study of people trying to interact with a new prototype photocopier system revealed that one of the fundamental problems in human-machine communication was that the human only has available the machine's instructions but does not know the internal logic of the machine, while the machine only has available to it the direct actions of the human upon it but does not have any means to understand the wider context of the human action.

In part, the drive to design context aware computing arose as a response to the identification of this fundamental communication problem, of both the machine needing to better understand the human and the human being needing to better understand the machine (mutual intelligibility).

But the possibility that mutual intelligibility can be fully addressed by making available more of the context to both the human and the machine, through sensing technologies, visualization, machine intelligence in the form of a more

extensive internal logic, and new representations has serious limits (Suchman, 1987; Dourish, 2004; Brown and Randall, 2004). And this poses limits and design paths for context aware computing.

Suchman's study established, drawing upon the traditions of ethnomethodology and conversation analysis, that how a human would or machine should act in a situation could not be completely specified a priori. Instead, while plans for action were helpful resources they could never be entirely complete, anticipating every possible situation and contingency. Suchman articulated that all actions are 'situated', that is - they depend for their meaning on the situation in which they occur. Actions are ad hoc, taken for specific rather than general purposes, and their meaning is irreducibly connected to the viewpoints, interactions, histories and local resources of those making sense of the situation at the time. (Harrison et al, 2006). While, plans are representations of action, actions themselves result not from rules or plans but from situations.

Extending the line of argument to context-aware computing, Dourish (2004) articulated that context cannot be seen as a form of information nor as stable, delineable and separate from activity. The scope of context that is relevant is defined dynamically, is particular to each occasion or activity and arises out of the activity itself. Moreover, as technologists our concern is to support the evolution of practice, the conversation with materials (Schon, 1983) out of which emerges new forms of action and meaning.

Brown and Randall demonstrated the importance of situated activity and the limits of predictive scenarios in trying to guide design when they considered the problem of the context aware telephone that knew when not to ring. The scenario posed seemed reasonable enough, that a phone will always vibrate and never beep in a concert, if the system can know the location of the cell phone and the concert schedule. But Brown and Randall's observation of how people decided whether or not a phone should be answered in everyday situations found that the phone was enmeshed in complex social contexts. People had appropriated caller ID, voicemail, or the help of friends to screen or avoid answering depending upon the situation at hand and had sometimes then decided to answer if the caller persisted, because it might be important and so on.

Brown and Randall's investigation led to design recommendations for context aware computing, that

(i) Technology provide context to users through simple structures (such as. caller ID) allowing users

to make sense of that contextual information themselves.

(ii) Context be used defensively, in such a way that incorrect inferences will not be a serious inconvenience to users.

(iii) Technology focus on communicating context, rather than attempting to compute and process it.

Further Brown and Randall (2004) emphasised the importance of dwelling with technology because conditions of possibility change and new norms can arise over time.

The ideas were further elaborated by (Chalmers and Galani, 2004), who introduced the notion of seamfulness, where interesting aspects of context (such as caller ID) are made explicit for human interpretation, as opposed to seamlessness, the promise of automatic systems, which inevitably fail to deliver in many circumstances.

Reasoning from the situated and interactional perspective has not halted the research and development of context aware computing! In many cases the context relevant to the application may be stable enough and straightforward enough that the functionality of the application is useful. For example, mobile phones with embedded GPS enable you to see your own location as a dot on a map, a very simple contextual structure that is also extremely useful, and while it does break down at times due to poor reception, that lack of movement of the dot on the map provides a clue that this is the case. Barkhuus and Dey, (2003) in limited studies, found that people were willing to accept a large degree of autonomy from simple context-aware applications as long as their usefulness was greater than the cost of limited control, and presumably the errors.

Recently and since the initial ethnomethodological critique of context aware computing, (Dourish, 2004) much of the research has tended toward the enabling middleware layer for context aware computing, establishing ways of dealing with complex context, sensor fusion, and inferring logic for context recognition. However, since many of the components needed for designing and building context-aware applications and services are now available in mobile phones (integrated GPS, accelerometers and other sensor functionalities), the time has come to revisit the design of context-aware services and applications that aim to closely support human activity. With the advent of social networks and the increasing interest in social context, context aware systems research is moving into more complex relations that involve more than one person, rather than just one person with their phone in the world.

3. SOCIAL AND MOBILE SYSTEMS AND THE PROBLEM OF MAKING THE IMPLICIT EXPLICIT

Kjeldskov and Paay (2010) considered the concept of indexicality another ethnomethodological concept, elaborated by Garfinkel and introduced to HCI by Suchman, as a founding and sensitizing concept for designing mobile context aware interfaces. Indexicality expresses the idea that our representations and actions often depend for their meaning on a reference to other things. Kjeldskov and Paay found “that people were particularly good at using visually prominent outlines of their immediate surroundings or distinct physical objects such as the layout of a room, parts of the skyline, or a distant tram) to align their mobile phone screen representation with their surroundings. They frequently used labels and headings in the system to match up with labels and signposts in their physical surroundings to create a meaningful indexical sign out of the information presented on the screen to that particular physical context. Their findings give recognition to the fact that context need only be hinted to and that much context can be implied through reference. However, in suggestions for further work they introduce the idea of systematic decomposition as the path toward elaboration of context aware computing.

“Further research also needs to extend the range of contextual factors indexed to, for example, the aspects of context related to activity, time and other information. This could also include a systematic decomposition of the different aspects of context and related sources of information that a system might provide an index to.”

This path of context elaboration and systematic decomposition puts context to the foreground (even though the thrust of Kjeldskov and Paay’s inquiry relates to subtle inference). The concern in putting context in the foreground and focusing system development in this way is that it tends to privilege that which is easily abstracted, while marginalizing other perspectives. It was only in evaluating the indexical mobile systems that the issue of privacy arose, from users, and it merits only a side-note.

“On a side note to this, once knowing how a system made use of people’s history and rhythm of social interactions, many people expressed concerns and uncertainty about how to control this system behavior in relation to issues of privacy.” Kjeldskov and Paay, 2010

The tension with respect to social context and privacy is evident. Social context seemed promising in the first instance as Kjeldskov and Paay had stated earlier on –

“Specifically, we found that people like to get an overview of their social context, such as the

presence and activities of other people in the surrounding environment.” “When social context was objectified it could be indexed to more successfully.” “In terms of the limitations of subtle factors of social context in the creation of meaningful indexical signs, representing social context in this way increases the potential for making interpretable indexical references to social context by taking something implicit and invisible and making it explicit and visible.”

The theme is that making social context explicit and visible increases its potential. And this is a dominant theme in current inquiry. The rapid uptake of social networking software certainly indicates that this is the case. However, it does leave aspects of non-use, the problems of explicit context such as privacy concerns, and the potential of implicit context as afterthoughts.

The use of explicit social context data made available in social networks that have little or no privacy controls (even when their users think that they do) leaves users susceptible to context aware spam and phishing attacks, which come from people who they think they know, using not only their email addresses but also their birthdays etc and other private information to generate authentic looking communications (Brown, Howe et al, 2009).

4. REFRAMING THE DESIGN OF CONTEXT-AWARE SYSTEMS

While the research question of how we can utilize context-aware computing is *prima facie* not an unreasonable one to ask, this paper argues that it may be more helpful to explore broader design challenges, within which context-aware computing might play a role, and to see within this framing, whether context-aware computing reveals itself as useful. Explored from a broader design frame, context-aware computing is not explored for its own sake, but at times when it is likely to be useful.

While the HCI community has somewhat embraced the ideas of ludic design, design for fun or pleasure, or critical design -- the notion that designs can provoke people to reflect (in the creative arts tradition), there has been little exploration within HCI of broader design challenges and broader design approaches in general. The dominant “design” approach within computer systems development is still seen as requirements elicitation (although this might be done now through ethnography) followed by systems development. Design methods tend to be adopted in the form of easy-to-apply techniques such as scenarios, or cultural probes. Where broader challenges are explored, the design attempts from the field of HCI often reduce quickly to ideas of explicit representation (although there are notable

exceptions in the design of tangible media). There is little room for ambiguity that might allow exploration and reframing.

This paper argues for a broader design framing for HCI research in general, challenging the requirements to system implementation paradigm. Design approaches are needed not only in areas that seek to explore particular value sets such as ludic design and critical design.

We introduce in this paper an investigation of context from within the broader framing of a design problem, the significant problem of solving urban congestion and pollution. A broader design investigation will ask why people travel, what motivates them, and is this the right frame and unit of analysis for the problem, or should we be looking in a different way to design futures that impact less.

We describe a situated approach to design below with the purpose of identifying some of its key themes. The design brief is to explore ways of reducing congestion and pollution and we focus here on an initial and loose proposition of designing a form of agile ridesharing socio-technical system.

The concept of agile or dynamic ridesharing is based on the premise that *mobile social software* could significantly ease logistical problems and provide improved convenience and usability of ridesharing by allowing people to easily contact potential ride-sharers in their extended ride-share social network in real time through mobile phones to arrange ad hoc rides. This premise is supported by research into ridesharing systems and cultures (Brereton et al, 2009). However, it remains an open question of how to design a successful system that encourages sharing while providing necessary privacy protection, fitting easily into people's daily lives and makes people feel comfortable about sharing. These are all aspects that need design exploration. In a true design exploration a larger exploration of the problem allows proposing and critiquing solutions and reframing the problem in Schon's reflective approach of see-move-see.

An assumption of most technology supported ridesharing systems is that a significant role of the technology support is to provide automatic ride matching by matching rider to driver based upon origin, destination and travel times. From an information systems perspective, the power of information technology is to provide this kind of automatic data matching, so that a system can efficiently bring together people. However, as acknowledged by all rideshare system providers, aspects of privacy, safety, incentives, personal preference, ridesharing community building, local geography and physicality all need to be addressed.

In questioning the conventional problem-solution framework, our prototyping approach set out to explore how people might want to communicate about ridesharing, while trying to make as few assumptions as possible about ways in which matching, community building, privacy and cost sharing might be addressed? See Brereton and Ghelawat (2010) and Ghelawat et al (2010).

Design as a situated process begins with fieldwork, design propositions and investigatory prototypes or probes to support fieldwork. In our case, the main probe was a simple investigatory communication prototype that allowed people to exchange travel messages with friends on their mobile phone or through web console, or to represent their ride in formal fields with information such as preferred start time, destination etc. The prototype allowed us to see what kinds of messages people sent in the moment of travel. A more detailed account of the general method of embedded, evolutionary, exploratory prototypes is found in (Heyer and Brereton, 2010).

5. LESSONS FOR CONTEXT_AWARE COMPUTING FROM SITUATED DESIGN

We draw lessons for context aware computing from fieldwork and observations of use of the prototype simple agile ride messaging system. An example message read:

A at 6:55am: Morning walk in - very flexible with start time. First meeting at 10am."

Keeping context implicit

A striking characteristic of the communication was that people often only gave as much specificity as they felt was needed to open a negotiation about sharing. People either (a) knew that others knew where they lived, so didn't need to give specific information, (b) were happy to make a small detour in order to share such that suburb level specificity was sufficient, or (c) were reluctant to give specific information in a general post, but happy to share in follow up private messaging during ride negotiation. (Brereton and Ghelawat, 2010)

The contingencies that people communicated were clearly kinds of situated knowledge embodied in situated actors, knowledge that would often not arise prior to the situation. Much communication could not be easily represented in formal fields and participants quickly eschewed the formal fields for text messages - (although there is a possibility that formal fields could evolve to be more useful shortcuts over time and with growth in participation).

B at 9pm Thurs: Child drop off at Dunmore at 8:50am Friday then to GP to meet Fred at

9:30am. Or Fred, I could meet you in church hill or dunmore?"

While many such messages were written in text, some more complex situations were only reported retrospectively in fieldwork.

I didn't want to park at the meters in the sun because I was on the way to cello lessons. The cellos were in the back of the car and it was a very hot day, but there was no alternative. Then I thought I probably could park in the sun if I wound down the windows a little bit and parked for less than 45 minutes, keeping my meeting short. That's what I did, and I was in such a rush I didn't enter that one into the system.

Reframing the design problem

People talked about meeting, walking, and sharing ferry rides, just as much as about offering lifts in vehicles, suggesting that the design brief be reframed from an agile ridesharing system to a socio-technical system to support meeting and travelling. Because one can be a legitimate peripheral participant and gain value from participating, even if one has no ride to share, a meeting and travelling message system could encourage broad participation, confronting the problem of growing to a critical mass of participants needed to make the system useful.

Privacy, profiling and interaction considerations were paramount to people in determining whether they would use such a system scaled to a larger group. Participants felt that awareness of other potential riders outside of ones friendship group was critical for adding value and motivation to use a ridesharing system. However designing ways to assess whether other riders were trusted is a non-trivial problem. (Ghelawat, Radke et al, 2010).

Reframing the role of the context-aware system

Taking a situated design approach then, *the focus of the design inquiry rests on investigating and exploring the aspects that are most critical in designing a socio-technical system that people will use in the long term and that fits with their habits. This is a quite different approach to explicitly representing context and seeing what issues that raises.* Moreover the role of a context aware system becomes defined as something that assists an action agenda – in this case, helping to make rideshare connections and opportunities while protecting privacy. Through an action agenda, real scenarios arise that can help reveal how context representation can support activity. In many cases, people did not want their locations revealed, but they were willing to show their GPS location in certain situations. For example, in one case, once a driver had negotiated to pick up a passenger, GPS representation would have helped the passenger to see how close the driver was, because the driver,

while driving, was unable to phone and give an update on how delayed they were. The rider couldn't relax and read their newspaper, because they had to keep looking out for the car in busy traffic to be able to hop in at a moments notice. In this kind of specific circumstance, once they had negotiated to share a ride, people were happy to have their GPS location displayed to selected people. Otherwise, people raised concerns of home robbery, surveillance etc. Context is occasioned and particular, as Dourish said in 2004, and so might its explicit representation be.

Creating broad historic context

While the temptation is to try to support personal representation of dynamic context, our investigation suggests that in this case, a promising approach is to create context that is historic, unchanging (by virtue of being in the past), and sufficiently anonymised that it does not identify individuals. In the case of growing ridesharing, a good visual representation showing the potential for rides from your home suburb, by virtue of anonymised past offers and approximate trips, could provide motivation to make personal connections and offer or accept rides oneself. That is, the act of making visible, something that is hidden, need not only apply to real-time situations with identified people. Separation is kept between what is personal and dynamic and what can be abstracted, anonymised and recounted as history. The concept is employed successfully in the online selling of books, where the system tells customers that people who bought this book of interest also bought these books.

6. DESIGN AND THREE PARADIGMS OF HCI

Harrison et al (2006) give lengthy consideration to the relation between design and three paradigms that can be identified in HCI research. The first paradigm consists of a-theoretic and entirely pragmatic approaches to HCI design. The second paradigm is organized around a central metaphor of mind and computer as coupled information processors and in this paradigm use-context is a source of information, which can be formalized and transmitted in service of design. The third paradigm recognizes situated perspectives and identifies that while meaning derives from information it cannot be summed up by mapping information flow: It is instead "irreducibly connected to the viewpoints, interactions, histories and local resources available to those making sense of the interface" and therefore to some extent beyond the reach of formalisation. Design that adopts a third paradigm research stance moves to studying the local situated practices of users, allows multiple interpretations and designs interfaces to fit their intended physical and social setting. Rather than seeing theory as primary, (such as context-aware

computing, if we consider that as a theory) and design as a way of instantiating and testing theory, understanding in this paradigm emerges from a combination of theoretical lenses and what happens practically at the scene of action – what Gaver has called “humble theory” (2006).

Combining a third paradigm situated approach with a design action agenda, a first paradigm pragmatic approach, leads us towards ways to devise the most effective communication system and “dashboard” for travel sharing that both tracks and motivates use by merging both implicit or tacit and explicit context. The overall approach involves immersion into the problem space to explore possibilities and possible reframing of the problem. It involves cyclic, iterative development in order to understand issues revealed over time in response to use as a system is deployed (Heyer and Brereton, 2010). The approach eschews an apriori assumption of making implicit context explicit in a second paradigm agenda. Instead combinations of tacit, visual and explicit expression are explored in situ, toward achieving uptake of travel sharing. Context-awareness is considered alongside and integrated with other approaches such as ethnographic study, recommender systems, visual design, persuasive computing, data mining, social software, industrial design and local community, policy and geographic considerations.

Rather than a conservative account of design that sees design solely as a problem solving activity, a situated approach to design, where understanding or construction of the situation is core, emphasises creative research into the situated nature of the problem space, *with theories deriving from design investigation as much as contributing to it.*

In this paper then we argue for a design approach to context aware-computing (and HCI research in general) that (i) undertakes an action agenda to address significant design problems, (ii) takes a creative situated approach to understanding human needs drawing in multiple theories as needed, and (iii) tests interfaces in real contexts of use. These are the real tests for the validity of our theories and the value of our interfaces.

7. ACKNOWLEDGMENTS

We are very grateful to participants in the rideshare project and to the Australian Research Council and taxpayer for Discovery grant DP110100530.

8. REFERENCES

Barkhuus, I. and A. Dey, (2003) Is context-aware computing taking control away from the user? Three levels of interactivity examined. *UbiComp* p.1-9

Brereton, M (2009) Designing from Somewhere: A located, relational and transformative view of design. In *(Re) searching a digital Bauhaus*. Jonas Löwgren, Lone Malmberg & Thomas Binder (Eds). Springer-Verlag London. Pp 99-119

Brereton, M. and S. Ghelawat, (2010) Designing for participation in local social ridesharing networks: grass roots prototyping of IT systems, *Participatory Design Conference*. ACM: Sydney, p. 199-202.

Brown, G., Howe, T., M. Ihbe, A. Prakash, and K. Borders, (2008) Social Networks and Context-Aware Spam, *Proc. CSCW*, San Diego p. 403-412

Brown, B. and R. Randell, (2004) Building a Context Sensitive Telephone: Some Hopes and Pitfalls for Context Sensitive Computing. *Comput. Supported Coop. Work*, 13(3-4): p. 329-345.

Chalmers, M. & Galani, A. (2004) Seamless interweaving: heterogeneity in the theory and design of interactive systems. *Proc. Designing interactive systems* ACM p. 243 - 252.

Dey, A., (2001) Understanding and Using Context, *Personal and Ubiquitous Computing*, Volume 5 Issue 1, p. 4-7.

Dourish, P., (2004) What we talk about when we talk about context. *Personal Ubiquitous Computing*, 2004. 8(1): p. 19-30.

Gaver, W. (2006). “Learning from experience”. *CHI 2006 Workshop Theory & Methods for Experience Centered Design*. <http://www.users.cs.york.ac.uk/~mblythe/CHI/Gaver.pdf>

Ghelawat, S, K. Radke and M. Brereton, (2010) Interaction, Privacy & Profiling Considerations in Local Mobile Social Software: Prototype Agile Ride Share, in *Proc. OZCHI* ACM: Brisbane. p. 376-379.

Häkkinen, J., A. Schmidt, J. Mäntyjärvi, A. Sahami, P. Åkerman, A. Dey (2009) Context-Aware Mobile Media and Social Networks MobileHCI'09: Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services p.1-3

Harrison, S., Tatar, D., and Sengers, P. (2006) The three paradigms of HCI. In *Alt. chi*, Proc CHI '06. ACM Press p.1-21

Heyer, C. & Brereton, M. (2010) Design from the Everyday: Continuously evolving, embedded exploratory prototypes, ACM SIGCHI Conference on Designing Interactive Systems, Held: Aarhus, Denmark Aug 16-20, pp1-10

Kjeldskov, J. and J. Paay, (2010) Indexicality: Understanding mobile human-computer interaction in context. *ACM Trans. Computer-Human. Interaction*. 17(4): p. 1-28.

Schilit, B., Theimer, (1994) M. Disseminating Active Map Information to Mobile Hosts. *IEEE Network*, 8(5). pp 22-32.

Schön, D.1983. *The Reflective Practitioner: How Professionals Think in Action*. New York: Basic Books.

Suchman, L. (1987). *Plans and Situated Actions*. Cambridge: Cambridge University Press.