Pers Ubiquit Comput (2004) 8: 19–30
DOI 10.1007/s00779-003-0253-8

ORIGINAL ARTICLE

Paul Dourish

What we talk about when we talk about context

Received: 16 June 2003 / Accepted: 30 September 2003 / Published online: 5 December 2003

Abstract The emergence of ubiquitous computing as a new design paradigm poses significant challenges for human-computer interaction (HCI) and interaction design. Traditionally, HCI has taken place within a constrained and well-understood domain of experience—single users sitting at desks and interacting with conventionally-designed computers employing screens, keyboards and mice for interaction. New opportunities have engendered considerable interest in "context-aware computing"—computational systems that can sense and respond to aspects of the settings in which they are used. However, considerable confusion surrounds the notion of "context"—what it means, what it includes and what role it plays in interactive systems. This paper suggests that the representational stance implied by conventional interpretations of "context" misinterprets the role of context in everyday human activity, and proposes an alternative model that suggests different directions for design.

Keywords Context-aware computing · Ethnomethodology

1 Introduction

One of the major research directions for human-computer interaction (HCI) over the past few years has been exploring the novel forms of interaction that can be achieved by integrating computer technology with the everyday physical world in which we live and work. This line of research goes by a number of names—ubiquitous computing (Weiser 1991), context-aware computing (Dey et al. 2001), pervasive computing (Ark and Selker 1999), embodied interaction (Dourish 2001), and more. Although the nomenclature varies, the central ideas are largely the same. Extrapolating from current trends in the development of low-cost and low-power devices, ubiquitous computing proposes a digital future in which computation is embedded into the fabric of the world around us. In this world, our primary experience of computation is not with a traditional desktop computer, but rather with a range of computationally-enhanced devices—pieces of paper, pens, walls, books, hammers, etc. The opportunity implied by this ubiquitous computing vision is to capitalise on our familiarity, skill and experience in dealing with the everyday world around us. The world can become an interface to computation, and computation can become an adjunct to everyday interaction.

There are many significant research issues that this vision encompasses, but two have become particularly prominent in HCI. The first is the mutual relationship between physical form and activity; how we can design computationally-enhanced devices and how their form as much as their interactive ability affects likely patterns of action and interaction. Researchers have looked towards other design disciplines to better understand the relationship between form and function and to explore the permeable boundary between physical and digital systems (e.g., Strong and Gaver 1996; Brave and Dahley 1997). The second concern, which will be the focus of the discussion here, is how computation can be made sensitive and responsive to its setting. How can sensor technologies allow computational systems to be sensitive to the settings in which they are used, so that, as we move from one physical or social setting to another, our computational devices can be attuned to these variations?

Whether we refer to it as context-aware computing or by one of its other names, the notion of "context" plays a central role in this new area of investigation. Context, in one form or another, has been a concern in many areas of design and computer science, but it is of central importance here. One reason is straightforward: when computation is moved "off the desktop," then we

P. Dourish (✉)
School of Information and Computer Science,
University of California Irvine, Irvine,
CA 92697–3425, USA
E-mail: jpd@ics.uci.edu
URL: http://www.ics.uci.edu/~jpd

suddenly need to keep track of where it has gone. The situation in which technology is used has become more variable, and so we need to understand more about it. So, a primary concern in ubiquitous computing research is to understand the potential relationship between computation and the context in which it is embedded.

Uses of context vary. Broadly, context is used in two ways in ubiquitous computing systems. In the first category, some systems encode context along with information so that it can later be used as a retrieval cue. Examples of systems that use context in this way include Lifestreams (Freeman and Gelernter 1996), Placeless Documents (Dourish et al. 2000), Where Were We (Minneman et al. 1995), and Forget-me-not (Lamming and Flynn 1994). A second, more common approach is to use context dynamically to tailor the behaviour of the system or its response to patterns of use. Examples of this sort of approach include Easy Living (Brumitt et al. 2000) and the Electronic Tourist Guide (Cheverst et al. 2000). Some systems, such as the Conference Assistant (Dey et al. 2001) combine both. (In restricting myself here to ubiquitous computing systems, I am passing over many uses of context in other domains such as information retrieval, natural language processing, multimodal interaction, etc.)

1.1 Context and social settings

The way in which the notion of "context" has entered the realm of computational design has largely been in the form of a set of design challenges—sensor fusion, information management, system parameterisation, and so forth. At the same time, however, designers have hoped that by incorporating notions of context into interactive technologies, those technologies can be made more sensitive to the details of specific settings of use, which social scientists have often pointed out in critiques of conventional system design (Suchman 1987). Social scientists have argued that traditional interactive system design often rigidly fails to respond to the setting in which action unfolds; by incorporating context, system designers have hoped to make their systems more responsive to the different social settings in which they might be used. The goal of incorporating these understandings into the design of technical systems has been explicitly avowed in the development of ubiquitous computing systems. For example, in an early article, Weiser comments:

> The idea of ubiquitous computing first arose from contemplating the place of today's computer in actual activities of everyday life. In particular, anthropological studies of work life teach us that people primarily work in a world of shared situations and unexamined technological skills (Weiser 1993).

Weiser cites the work of Lucy Suchman and Jean Lave as examples of this approach, and Suchman's notion of "situated actions" is a common source for the idea that computer systems should respond to the settings within which they are used. Abowd et al. (2002) similarly cite Suchman's work as motivating this concern:

> Situated action emphasises the improvisational aspects of human behaviour and deemphasises a priori plans that the person simply executes... Ubicomp's efforts informed by a situation action also emphasise improvisational behaviour and would not require, or anticipate, the user to follow a predefined script (Abowd et al. 2002).

However, the social and technical ideas often sit uneasily together. Ubiquitous computing systems may be more responsive, and yet they seem to fail to address the sociological critique. Turning social observation into technical design seems to be problematic. The goal of this paper is to explore this relationship between technical and social aspects, and to examine the mismatch between the two. In particular, I want to do this by discussing how the central issue of context appears from each perspective. To paraphrase Raymond Carver, what do we talk about when we talk about context?

2 Two views of context

As I have suggested, the notion of context in ubiquitous computing has a dual origin. On the one hand, it is a technical notion, one that offers system developers new ways to conceptualise human action and the relationship between that action and computational systems to support it. On the other hand, it is also a notion drawn from social science, drawing analytic attention to certain aspects of social settings. Translating ideas between different intellectual domains can be both exceptionally valuable and unexpectedly difficult. One reason is that the ideas need to be understood within the intellectual frames that give them meaning, and we need to be sensitive to the problems of translation between these frames.

2.1 Positivist and phenomenological reasoning

Broadly, theories in the social sciences fall into three broad categories: positivist, phenomenological, and critical.

Positivist theories derive from the rational, empirical, scientific tradition. By analogy with the way that physical scientific theories seek to reduce complex observable phenomena to underlying idealised mathematical descriptions, positivist theories seek to reduce social phenomena to essences or simplified models that capture underlying patterns. Accordingly, positivist theories seek objective, independent descriptions of social phenomena, abstracting from the detail of particular occasions or settings, often in favour of broad statistical trends and idealised models. Positivist theories are often

(although not always) quantitative or mathematical in nature.

In contrast to the objective and quantitative nature of positivist theories, *phenomenological* theories are subjective and qualitative in orientation. By "subjective" I mean that they regard social facts as having no objective reality beyond the ability of individuals and groups to recognise and orient towards them; in this view, social facts are emergent properties of interactions, not pregiven or absolute but negotiated, contested and subject to continual processes of interpretation and reinterpretation. Phenomenology turns analytic attention away from the idea of a stable external world that is unproblematically recognised by all, and towards the idea of that the world, as we perceive it, is essentially a consensus of interpretation. Phenomenological theories argue that abstract categories, for instance, are things that need to be imposed on the world through our interactions with it and with each other, rather than things that exist within it.

Critical theories tend to have a broader scope. Critical theory, particularly associated with the Frankfurt School and theorists like Adorno, Horkheimer, Marcuse and Foucault, is essentially an extension of Marxist analysis. The basis of Marxist historical materialism is that social and economic conditions are the outcome of a historical process of evolution and reflect an evolving, dynamic (and generally unequal) balance of power and control between social groups. Essentially, the nature of human existence is a product of social and economic conditions which themselves reflect the historical distribution of power and control in society. Critical theory extends historical materialism beyond the social and economic "products" of society and to its intellectual products too, arguing that ideas, language, and modes of thinking are similarly conditioned.

While I include critical theories here for completeness, it is the distinction between positivist and phenomenological theories that is relevant for this paper. The relevance of this distinction is that engineering approaches—including those that tend to dominate discourse about ubiquitous computing—inherit a positivist tradition, while many approaches to social analysis relevant to HCI design—including the ethnomethodological position practiced by Suchman and cited by Weiser—are heir to a phenomenological legacy.[1] Where positivist approaches posit accounts of social life that are independent of the observer, phenomenological theories note that agency and interpretation are the central facets of all social action—including the social action of theorising about social action. The analytic constructs of positivist accounts of social action are, themselves, the outcomes of social action, and so inherently cannot be separated or made independent of sociality as positivism claims. Consequently, these two positions are incompatible.

By the same argument, the concepts of "context" put forward by the positivist engineering tradition and the phenomenological social tradition are similarly incompatible. Suchman's argument concerning the situated nature of action does not merely observe an otherwise unnoted aspect of how people go about carrying out tasks; instead, it is a critique of positivist accounts of human social action. As a consequence, the subsequent incorporation of "context" into system design is something to be examined more closely.

2.2 Context as a representational problem

We can think of the positivist account of context as defining the problem as one of representation.

Software systems are representational, so a concern with context naturally leads to a concern with how context can be encoded and represented. For instance, Schilit and Theimer (1994) define context as "location and the identity of nearby people and objects." Ryan et al. (1997) define context as "location, identity, environment, and time." In one of the more extensive investigations of context-based computing, Dey et al. (2001) define context as "any information that can be used to characterise the situation of entities" and elaborate it as "typically the location, identity and state of people, groups, and computational and physical objects." One of the broadest definitions is one of the earliest; Schilit et al. (1994) observe:

> Context encompasses more than just the user's location, because other things of interest are also mobile and changing. Context includes lighting, noise level, network connectivity, communication costs, communication bandwidth and even the social situation, e.g., whether you are with your manager or with a co-worker.

Again, my concern here is not to compare and critique these specific definitions, but rather to explore the "conceptual work" that context is doing here. In particular, I am concerned with the assumptions and commitments that are reflected in these definitions. In particular, four assumptions seem to underlie the notion of "context" as it operates in these systems.

- Firstly, *context is a form of information*. It is something that can be known (and hence encoded and represented much as other information is encoded and represented in software systems).
- Secondly, *context is delineable*. We can, for some set of applications or application requirements, define what counts as the context of activities that the application supports, and do so in advance.
- Thirdly, *context is stable*. Although the precise elements of a context representation might vary from application to application, they do not vary from

[1] Although I described the three approaches as characterising theories of social science, they also apply more broadly, and engineering approaches to HCI inherently embody (perhaps implicit) theories of social analysis and social action.

instance to instance of an activity or an event. The determination of the relevance of any potential contextual element can be made once and for all.
– Fourthly, and most importantly, *context and activity are separable*. Activity happens ''within'' a context. The context describes features of the environment within which the activity takes place, but which are separate from the activity itself. So, I might be engaged in a conversation, which may be happening in a location; the conversation is my activity, while the location is an aspect of the context. Broadly, although the ubiquitous computing program wants to observe that there may be fruitful relationships to be defined between context and content, they can be defined and described separately; the content or activity is ''within'' while the context is ''without.''

Independently of any particular definition of context for any particular application or set of applications, these four assumptions underwrite the notion of context as it appears in much current ubiquitous computing research. In particular, the idea that context consists of a set of features of the environment surrounding generic activities, and that these features can be encoded and made available to a software system alongside an encoding of the activity itself, is a common assumption in many systems. It is inherent in the notion that our systems will ''capture'', ''represent'' or ''model'' context—the normal and appropriate concerns of positivist design.

2.3 Context as an interactional problem

However, as noted earlier, although the drive to represent context is inspired by (and in some ways a direct response to) sociological investigations of real-world practice, those investigations come from a quite different intellectual tradition. This leaves us, potentially, in the awkward situation of attempting to derive positivist responses to phenomenological arguments—when the true target of those arguments is not just specific definitions of plans, actions and contexts, but rather the conceptual status of positivist models and representations. Essentially, the sociological critique is that the kind of thing that can be modelled, using the four principles mentioned above, is not the kind of thing that context is.

This is not the place, of course, for a full exploration of the relative merits of positivist and phenomenological accounts of the social world. Rather, in the rest of this paper, I want to focus on an alternative view of context, and discuss some of its implications. This alternative view takes a different stance of each of the four assumptions mentioned above:

– Firstly, rather than considering context to be information, it instead argues that *contextuality is a relational property* that holds between objects or activities. It is not simply the case that something is or is not context; rather, it may or may not be contextually relevant to some particular activity.
– Secondly, rather than considering that context can be delineated and defined in advance, the alternative view argues that *the scope of contextual features is defined dynamically*.
– Thirdly, rather than considering that context is stable, it instead argues that context is particular to each occasion of activity or action. *Context is an occasioned property*, relevant to particular settings, particular instances of action and particular parties to that action.
– Fourthly, rather than taking context and content to be two separable entities, it instead argues that *context arises from the activity*. Context isn't just ''there'', but is actively produced, maintained and enacted in the course of the activity at hand.

In other words, what I want to do here is to reconsider context, not as a representational problem but as an interactional problem. As a representational problem, the central concern with context is with the questions, ''what is context and how can it be encoded?'' What I want to argue here, by turning to social science investigations of everyday activity, is that this representational stance reflects a misunderstanding of the nature and role of contextuality in actual everyday affairs. Instead, I want to propose an interactional model of context, in which the central concern with context is with the questions, ''how and why, in the course of their interactions, do people achieve and maintain a mutual understanding of the context for their actions?''

This distinction reflects the essential difference between the positivist and the phenomenological positions. From a positivist perspective, context is a stable feature of the world, independent of the actions of individuals. From a phenomenological perspective, contextuality comes about only when it is mutually recognised by the parties to some interaction, drawing on their everyday, cultural, common-sense understandings of the nature of the social world. So, in this model, context isn't something that describes a setting; it's something that people do. It is an achievement, rather than an observation; an outcome, rather than a premise.

As an analogy, consider the notion of ''relevance'' in conversation. We can think of what people generally mean by context as denoting a kind of middling relevance to the matter at hand. For two people holding a conversation about a book they lately read, certain topics—such as their immediate subject of conversation, say the author of the book—are very relevant, while certain others—such as the scorer of the winning goal in the 1953 British F.A. Cup Final—are irrelevant (unless the book is about sporting trivia, of course). We would not normally describe either of these as the ''context'' for the conversation. Instead, the matters that we might designate as being contextually relevant are ones that may have a general bearing on the conversation and assist in its flow and construction—such as these conversants' shared history of interaction, the anticipation of an upcoming visit of the author to a

local bookstore, or their appreciation of the author's other work. These are issues of intermediate relevance—neither directly relevant nor directly irrelevant. So, similarly, the typical conception of context in technical systems is of information of a middling relevance—where an action takes place, the people at hand, and so forth.

The essential feature of this issue of relevance, though, is that the relevance of a certain matter to the conversation at hand will emerge from the conversation as it proceeds. The participants may change the subject or otherwise turn a matter from one of middling relevance to central relevance, or vice versa. They may suddenly realise an assumption that doesn't hold and need to backtrack to repair the conversation. They might turn the location of the conversation from "context" to "content" by remarking that it's too public a place and perhaps they should move elsewhere; and so forth. In other words, the determination of relevance—or of contextuality—is not one that can be made a priori. It is an emergent feature of the interaction, determined in the moment and in the doing.

This suggests that, context and content (or activity) cannot be separated. Context cannot be a stable, external description of the setting in which activity arises. Instead, it arises from and is sustained by the activity itself.

To provide some background to this, the next section will discuss some contributions from a particular analytic approach to social action which is centrally concerned with the detailed structure of sequences of action. On the basis of this, I will propose an alterative conception of context in ubiquitous computing and some design considerations.

3 Ethnomethodology

The particular approach to the analysis of social conduct that I will draw on here is ethnomethodology. Ethnomethodology is firmly grounded in the phenomenological tradition, as well as being an approach which is directly related to the motivations behind the move from traditional system design to ubiquitous computing, as detailed earlier. A detailed explanation of what ethnomethodology is (and why) is not appropriate here, but let me provide a sketch in very broad strokes (for more information, see Dourish and Button 1998; Heritage 1984; Garfinkel 1967).

Arguably, the central problem of social science is the "problem of social order": given that social action arises from the independent action of many individuals, how is it that it can be found, by those individuals, to be orderly, rational, or meaningful? Action in the world, at all levels, from conversations to courtroom proceedings to political campaigns, are understandable and meaningful activities, recognisably orderly in the way they proceed, even though they are outcomes of the separate and individual activities of individuals who have access only to their own mental lives. How does this work?

Broadly, we can conceive of two sources for this orderliness—orderliness from without and orderliness from within.

The first approach suggests that the orderliness of any particular instance of social action derives from a set of rules, expectations, norms and conventions that have some broad reality outside the particular occasion of the activity. So, for instance, our society might embody a set of rules about how particular forms of interaction, such as between a professor and a student, should proceed, based on a set of concepts such as power, class, dominance, deference, etc. These concepts, which we internalise in a process of socialisation, lead to a set of dispositions to act in one way or another, which in turn shape or mould specific occasions of action; it is the role of social science to recover the formal analytic categories that operate "behind the scenes" of daily life.

Ethnomethodology makes two comments about these kinds of formal analytic categories. First, it notes that people are not "judgemental dopes," whose actions are determined by unknown and abstract rules of this sort; those rules and formal categories are valid and relevant only to the degree that people enact them, point to them, adopt them and demonstrate them to each other. The shared reality of a rule is a *consequence,* not a *cause,* of the fact that people choose to follow it. Second, following Wittgenstein, ethnomethodology observes that the meaning of linguistic terms arises from social understandings of appropriate use rather than from external, objective reality; the word "hello" can be not only a greeting, but an inquiry, a rebuff, a joke, an exclamation, etc., depending on how, when and by whom it is used. Similarly, formal analytic concepts such as power, class, dominance, etc., are themselves linguistic terms whose stability is a matter of appropriate use not of external reality.

By contrast, then, the second approach to the source of orderliness of social action suggests that it arises from within the action itself. It is not imposed from outside, by the action of broad rules of conduct or social norms. Instead, it is an achievement of social actors. Parties to some activity find, within the circumstances of action, the means to render that action meaningful and orderly; and, similarly, they make such resources available to others. In the course of a conversation, for example, participants construct their utterances so as to demonstrate that they are appropriate to the action at hand—so that they are demonstrably answers to questions, elaborations of points, interjections, agreements, questions, and so forth. Similarly, these demonstrations of appropriateness furnish to others the means to see and understand what is going on as being what is meant to be going on. So, orderliness or rationality is not an abstract property of activity in general, but rather is a property of action as enacted by specific individuals in specific circumstances. The social world is not orderly because people blindly follow social rules that are imposed on them. Instead, finding the social world orderly and meaningful is a practical problem that people solve,

endlessly and unproblematically, as they go about their business.

4 Being ordinary

Ethnomethodology is concerned with social action in general, and its approach to examining how orderliness of action arises from within has been applied to a wide range of forms of social action, from the conduct of science (Garfinkel et al. 1981) to crossing the street (Sudnow 1972). One subfield of ethnomethodology, conversation analysis, applies the ethnomethodological perspective to the analysis of naturally-occurring conversations. Conversation analysis seeks to uncover how the organisation and meaningfulness of a spoken conduct is achieved, in real time, by the conversational participants.

The founder of the field of conversation analysis was Harvey Sacks. Sacks analysed fragments of naturally-occurring conversation to uncover the unnoticed mechanisms by which people would manage the conversation as it proceeded. In his paper "On doing 'being ordinary"', Sacks (1984) explores the question of ordinary action in a way that, firstly, helps to make the description of ethnomethodology above a little more concrete, and, secondly, casts some light on an ethnomethodological view of context.

The topic of the paper is the ordinary character of everyday activity, the very fact that most things are ordinary and unremarkable. In particular, Sacks is concerned with ordinariness as a feature of conversation, addressing the question, "how is the ordinariness of activities displayed, understood, and achieved in the course of conversation?" Recalling the discussion of relevance earlier, it is clear that ordinariness is a crucial feature of everyday conversation. Any given utterance rests on an uncountable number of assumptions and background understandings that are part of the ordinary fabric of daily life for the speaker and hearer. So, when someone tells me, "I saw an accident on my way to the bank this morning," I hear that the accident is something out of the ordinary, something worth reporting. Crucially, however, I do not question that going to the bank is something that this person might do, that they generally engage in acts of going places, that accidents are things that might happen in the sorts of places one might be when going to banks, that mornings are a time that people might be about their business, that an accident is something that can be seen, etc. These are ordinary matters, and no explanation for them need be sought or provided. If it were not possible to designate these things as simply ordinary, conversation would be impossible. So the question of ordinariness is a central one for studies of conversation.

The main element of Sacks' analysis is that activities or features of action are not simply "ordinary" in their own right, but rather are designated as (and recognised as) ordinary by social actors in the course of their activities. So, for example, if we look at conversational speech, we can see that the ways in which specific topics or events are woven into the speech serves to mark them as being ordinary or unremarkable. Speech is organised so as to designate the distinctions between those things that are remarkable and those that are not. What people report in everyday speech are not simply states of affairs, but the ordinary nature of those states of affairs, and this is done by the way in which they are spoken about or not spoken about. Reporting on a scene witnessed in the street also reports that the witnessing of that scene is a normal, unremarkable, ordinary thing that a speaker has a right to have done. We mutually understand, and reinforce in the course of our interactions, what it is that ordinary people do in ordinary circumstances; and it is the mutual recognition of this very ordinariness that allows us to designate activities as being ordinary, unremarkable and expected.

More broadly, this applies not only to conversation but also to action. As Sacks observes, people are unlikely to say, "Hey, let's do something ordinary tonight"[2]; but nonetheless, what we do when we spend an ordinary evening is engage in activities that we know or anticipate that many other people are doing as ordinary courses of action. In both action and conversation, "being ordinary" is something that people work at, by acting in ways that they understand to be the normal activities of the groups of which they are members, and making no issue out of them in the course of their interactions. The ordinariness of everyday affairs is an accomplishment of social actors; unquestioned, unnoticed, but an accomplishment nonetheless.

Ordinariness, in addition, is relative to particular communities and sets of activities. The ordinary and mundane features of life for research scientists are quite different from those for bank robbers and astronauts. So, while astronauts might refer to or orient to occasions of Africa passing over their heads, and do so in ways that mark those as clearly and obviously ordinary courses of events, those same references would clearly not be ordinary things for other groups of people. Ordinariness, too, is relative to particular sets of circumstances; ordinary ways of acting on airplanes and at parties are clearly different.

So, three significant aspects of ordinariness emerge from this analysis. Firstly, ordinariness is something that we do; rather than simply being a stable feature of the world, it is actively managed and achieved in the course of interaction. Secondly, this is a mutual achievement; ordinariness must be both produced and recognised by the parties to an interaction. Thirdly, it is relative to particular communities and activities; it is a feature of forms of competent language use for groups of language users.

[2] Except, perhaps, in some ironic sense; which, of course, would not be ordinary at all. This is clearly not something that one would *ordinarily* say.

It is, no doubt, reasonably clear why Sacks' discussion of ordinariness is relevant here. *Context is an occasioned property of action in just the same way as ordinariness*. Just as ordinariness, or unremarkableness, or relevance of some utterance for the emerging conversation is an achievement of that conversation, so too is any distinction between an activity and its relevant context. Like ordinariness, context is managed moment by moment, achieved by those carrying out some activity together, and relative to that activity and to the forms of action and engagement that it entails. Sacks' discussion of ordinariness as an achievement of social actors illustrates what I mean by making a conceptual turn, from treating something as a "representational" problem to treating it as an "interactional" problem.

Having outlined this alternative model of context, I want to go on now to consider some of its consequences for interactive system design. To do this, I need to place things in a larger frame.

5 Practice

I have argued that context is a feature of interaction. The traditional concern in context-based systems is to be able to use the context in order to discriminate or elaborate the meaning of the user's activity. If activity is the site of contextual manipulation, then this move argues for a link between action and meaning as the primary concern of ubiquitous computing. We find the link between these two in the concept of practice.

In HCI and CSCW, we are familiar with the concern with work practice, as expressed by ethnographic and other "rich description" approaches to studies of working settings (Plowman et al. 1995; Luff et al. 2000). In these areas, the informal understanding of practice is "the detail of what people actually do" (in contrast to the formal procedures that organisational rulebooks might suggest that people ought to do.) The concern with designing software systems around the details of real needs and real use has led to an interest in techniques that can help uncover practice.

In his book *Communities of Practice*, Etienne Wenger puts forward a richer notion of practice. For Wenger, practice is not merely about what people do, but about what they experience in the doing. "Practice," he explains, "is first and foremost a process by which we can experience the world and our engagement with it as meaningful." (Wenger 1998)

Wenger's focus on meaning comes from his particular interest in learning within communities of practice. As an example, imagine the case of an apprentice shoemaker. On his first day on the job, he encounters a host of different pieces of leather arrayed around the workshop, but beyond basic characteristics like size, shape and colour, he has no way to discriminate between them. They are, by and large, all just pieces of leather; they simply do not mean very much to him. However, as he becomes more skilled, and gradually moves from the periphery to the centre of the community of practice, he slowly learns to see and understand the leather as a shoemaker sees it. His experiences and developing skills teach him new ways to assess, understand and encounter the leather. He begins to understand it as it relates to the practice of shoemaking. It takes on new meaning in terms of its durability, workability, weaknesses and the opportunities it holds for particular sorts of use in the hands of a skilled shoemaker. As a skilled practitioner, he encounters different pieces of leather in terms of their possibilities for his craft. It is not simply that his perspective has changed; rather, in learning a new form of practice, he has learned a new set of ways of finding (some part of) the world as meaningful. The same sort of argument applies beyond the obvious cases of physical skills. So, for example, the process of computer science education involves teaching students not just how computer systems work, but, critically, *how to see the world as computer scientists do*. For example, it teaches people a stance from which the world reveals itself to them as available for certain kinds of representation and automation. Students learn to see iteration, recursion and encapsulation in the settings around them (and will often begin to speak of them that way). In general, practice—the practice of being a computer scientist, or the practice of being a shoemaker—is about finding the world meaningful, in terms of the actions that it affords. As one acquires these skills, new aspects of the environment become relevant for the activities that one performs. The scope of potential context broadens.

So, the concept of practice is one that unites action and meaning. It describes how the world reveals itself to us as one that is meaningful for particular sorts of actions, and as a result of our participation in communities. As competent social actors in particular domains, we can find the world and the settings we encounter as meaningful. This unification of action and meaning is also central to the question of context, since context is essentially about the ways in which actions can be rendered as meaningful—how a particular action, for example, becomes meaningful for others by dint of where it was performed, when, or with whom. The questions I raised earlier—what constitutes context, how people may orient to features of the world as contextual or central, how relevance is managed, etc.—are questions of practice.

What is crucial to the interactional (rather than representational) view is to see practice as a dynamic process. It evolves and adapts. As technologists, then, our concern is not simply to support particular forms of practice, but to support the evolution of practice—the "conversation with materials" (Schön 1983) out of which emerges new forms of action and meaning.

This is no mean challenge. Nor is it a single, distinguishable goal. Rather, it is a broad agenda for research in interactive system design. As a result, then, no absolute solutions can be presented here. Instead, I will give

examples that serve to illustrate the potential design consequences of a primary concern with practice and evolution.

6 Practice and technology

As discussed above, the representational view of context arose as a response to practical problems on the part of system designers. So, if practice is the lens through which we can resolve the problems of context, then what does that imply for the design of new technical artefacts and systems?

There are two central elements to the argument that this paper has developed. Firstly, I have argued that context is an emergent property of occasions of interaction, rather than being a stable, objective set of features that externally characterise activity. Context remains critically important for understanding, contextualising and disambiguating forms of activity and information, but it is in the nature of context to be continually negotiated and redefined. Secondly, I have argued that these contextual properties take on their meaning or relevance through their relationship to forms of practice; that is, it is engaged action around artefacts and information that make those artefacts meaningful and relevant to people. The meaning of a technology, then, cannot be divorced from the ways that people have of using it. Critically, how technology is used in practice is not the same as the designer's conception, in two ways. Firstly, at a gross level, people often find ways of using technology that are unexpected or unanticipated; for example, the adoption patterns of most communication technologies, from telephones to electronic mail to SMS messaging, have taken on forms in use that surprised their designers or implementers (Fischer 1992; Sproull and Kiesler 1991; Ling and Yttri 2002). Secondly, at a more particular level, even when the general patterns of technology use do conform to expectations, the meaning of the technology for those who use it depends on how generic features are particularised, how conventions emerge, and so on (Mark 2002). Specifying a hierarchical storage system in abstract is one thing; understanding how specific groups of people will make use of particular categories is quite another (Dourish et al. 1999.)

What this suggests, then, is that the major design opportunity concerns not use of predefined context within a ubiquitous computing system, but rather how can ubiquitous computing support the process by which context is continually manifest, defined, negotiated and shared? Ubiquitous computing technologies extend the reach of computation into the everyday world, and that world is one in which, through our everyday practice, we enact, sustain and reproduce new forms of social meaning. The meaning itself may, by definition, be something that can never be removed from the social world and encoded in the technical. Nonetheless, though, technology plays a critical role in the evolution of meaning within communities of practice.

As illustrations, the three examples below draw on current work that embodies aspects of this approach.

6.1 Systems that display their context

This approach draws our attention to the way that users of computer systems need to form interpretations of the state of the machine in order to determine appropriate courses of action. Hutchins et al. (1986) identify the "gulf of interpretation"—the difficulty of interpreting the system's state as a response to the user's command—as one of the major problems that HCI design attempts to address, but this problem extends beyond simply the execution of commands. Especially in modern networked systems, matching action to the state of the system can be extremely problematic. Typical interfaces systematically hide the details of their operation, even though those details are frequently crucial to an evaluation of the current state of the system. Has the system paused because it's working, because of a network problem, or because of low memory?

One solution to this problem is to consider how a system can display aspects of its own context—its activity and the resources around which that activity is organised. Looking again at the issue of practice, the goal is to allow the system to reveal a richer picture of activities, and so provide users with a more nuanced interpretation of the meaning of the system's action (Dourish 1995).

As an example, one domain where we are currently exploring this approach is in that of the security of networked systems. Technical investigations of security have provided us with systems that can offer mathematically provable encryption, zero-knowledge proofs, etc. For users, though, the central question is, "is this system secure enough for the task I need to perform now?" Two things are striking about this question. Firstly, that "secure" and "secure enough" are quite different things; and secondly, that most interactive systems do not provide end users with the means to make a reasonable determination of the degree of security currently available.

Current work is investigating the extent to which visualisation techniques can provide users with a dynamic understanding of the security configuration of their system and its applications, and thus provide them with the resources to make these sorts of determinations (Dourish and Redmiles 2002). This is clearly neither a question of simply defining degrees of security, nor of prescribing secure and insecure operations, but, rather, allowing the user to make continual determinations of the potential consequences of their actions and their opportunities to reconfigure or realign the technologies through which they are conducting their actions.

One critical issue here is the dynamic nature of security and configuration information. This is especially important in mobile settings. Most laptops, for instance, are used to connect through many different

network arrangements, from the office, home, or on the road. Similarly, handheld devices move through environments in which they are handed off from one network infrastructure to another. The flipside of the "seamless access" to which most mobile systems aspire is the systematic hiding of network heterogeneity; but it is precisely that heterogeneity that determines users' assessment of security. While maintaining seamless access, our approach aims to visualise aspects of this heterogeneity, and so allow users to explore the technical context in which their applications operate.

6.2 Architectures for adaptation

One of the consequences of how information systems are incorporated into different settings and acquire different meanings in the course of their use is that they are adapted to the different settings in which they are used. A number of researchers have studied aspects of tailorability (how systems can be adjusted by their users to suit individual needs) and adaptation (how systems can infer patterns of actions of users and adapt themselves to them). Examples include the work of Mackay (1990), MacLean et al. (1990), and Mørch and Mehandjiev (2000). However, although they are strongly related, these two areas of work focus on different issues. Tailorable systems place control in the hands of users, but frequently offer control only over surface-level features of the system's behaviour, such as user interface structure and appearance, short cuts and defaults. In turn, adaptive systems may address more foundational issues (including component distribution, optimisation, etc.) but generally operate autonomously. One of our interests is in what sorts of infrastructures allow for user-initiated customisation of "deep" software structures.

In prior research, I have investigated the use of "reflective" architectures to support this sort of deep customisation (Dourish 1998). Current investigation is focused on less computationally expensive mechanisms for achieving similar ends, specifically for supporting "migratory" applications—ones that move from place to place and from device to device. Our infrastructure, called Emigré, uses run-time XML-based architectural descriptions (Dashofy et al. 2001) to allow applications to reconfigure themselves to local technical conditions (e.g., device and bandwidth availability) and patterns of use (e.g., synchronous or asynchronous interaction). The important point, again, is that the internal system structures that affect how the system can be used are available for some degree of user inspection and manipulation, and that this reflective activity be fairly seamlessly integrated with conventional use, by analogy with the way that everyday objects are available to use both as artefacts and tools simultaneously.

So, in this approach, the system's own structure and action become available as context for making decisions about adaptation and configuration of device resources.

6.3 Structures in information spaces

In information work, the meaningfulness of information for people's work is often encoded in the structures by which that information is organised. For example, Bowker and Star (1999) have provided a detailed exploration of the way that structures of classification and their use reflect and embody tacit understandings of the meaning of information for particular groups. Similarly, in some work on the use of classification structures in engineering work, we observed the ways that information structures in a single organisation were intimately tied to particular forms of use, causing problems for the mutual intelligibility of information (Dourish et al. 1999).

We are interested in the ways in which we can provide information workers with interfaces that give them a direct experience of the structures by which information is organised, especially in collaborative settings. Drawing on research into spatial hypertext (Marshall and Shipman 1993), we are building systems for collaborative sensemaking in which structures are directly accessible as part of an information-centric model of interaction. In traditional hypertext systems, such as on the Web, users organise information by creating explicit structures of links from one item to another. By contrast, spatial hypertext systems allow users to organise information in free-form two-dimensional space, much like layout cards on a table; by drawing on principles of human perception, the system analyses the layout of the information to find structures that are visually salient to users, such as clusters, piles, columns, tables, etc.

This style of interaction has a number of valuable properties, but three are especially relevant here. The first is that users interact directly with the information objects rather than with abstract structures by which those objects are organised. There is no separation between two levels of control. The second is that structure emerges in the course of a users' interaction, rather than having to be specified all at once or in advance of the actual data being incorporated. The third, in a collaborative setting, is that users negotiate the structures by which information will be organised, incrementally building on each other's work in creating a collective organisation for information that reflects their immediate needs, concerns and understandings.

As in the previous examples, this approach to information organisation sees the structure and meaning of interaction arising out of the detail of that interaction itself. The system is specifically designed to allow forms of practice to emerge and evolve, rather than requiring users to fit their work and their information into predefined patterns.

6.4 Summary

By turning our attention from "context" (as a set of descriptive features of settings) to "practice" (forms of

engagement with those settings), we assigned a central role to the meanings that people find in the world and the meanings of their actions there in terms of the consequences and interpretations of those actions for themselves and for others. The important point, however, is that we now see those meanings as essentially open-ended; we recognise that part of what people are doing when they adopt and adapt technologies, incorporating them into their own work, is creating and communicating new meanings though those technologies as their working practices evolve. The broad principle that these examples illustrate is that users, not designers, determine the meaning of the technologies that they use, through the ways in which they incorporate them into practice. Accordingly, the focus of the design is not simply "how can people get their work done," but "how can people create their own meanings and uses for the system in use"; and in turn, this suggests an open approach in which users are active participants in the emergence of ways of working.

It is critical to note that this implies more than simply a change to the ways in which we go about designing technologies, but also a change to the technologies that we design. It is not simply a change in process. While approaches such as participatory design (PD) have made significant methodological contributions to the ways in which we consider user involvement in the design process, the artefacts that emerge from PD are still stable, static, and closed software systems. Instead, the approach I have advocated here is one that, while compatible with the goals of PD, requires that we change our notion of software artefacts themselves and their relationship to practice.

Each example illustrates one way of responding to this. In the first, the relationship between context and action is turned around, so that systems display aspects of their own context (which is more fixed) for interpretation by end users. In the second, the focus of context is again technical rather than human, but here relationship between context (metalevel description) and content (the system action) is laid open as one that is continually maintained and available for inspection and change. In the third, the structure that makes information meaningful is allowed to emerge out of human activity, rather than being specified in advance. In all three, the relationship between form, practice and meaning is re-evaluated.

Although I discuss other related principles elsewhere (Dourish 2001), I do not provide formal, step-by-step design guidelines, for two reasons. Firstly, our understandings of ubiquitous computing technologies, their potential and their uses, are currently insufficiently developed to formalise design measures. Secondly, my goal here has not been explicitly design-oriented, but conceptual; to investigate the "category error" implicit in many discussions of context and human activity. However, the examples presented here do, I hope, serve to illustrate that the perspective I have outlined is not one that argues against the design of innovative technologies, but rather one that forces a reconsideration of the relative roles of human and technical agency in the design of ubiquitous and interactive systems.

7 Context and embodied action

What I have been presenting here is an alternative view of context in interaction. This is not simply a different technical model or a different design for how context can be represented. Rather, it is a different proposal about what context is. The difference is significant. In particular, it undermines the idea that, somehow, by encoding contextual information, interactive systems can be brought closer to the situated or embedded models suggested by social analysts (Suchman 1987). This project fails because it relies on a set of assumptions about context—the assumptions outlined in Sect. 2.2—that do not hold in practice.

In contrast, I have presented a model of context here in which context and activity are mutually constitutive. This approach is one that I have been calling "embodied interaction." The essential feature of embodied interaction is the idea, as illustrated above, of allowing users to negotiate and evolve systems of practice and meaning in the course of their interaction with information systems.

One feature of the examples that might seem unexpected is that they do not simply deal with the traditional domain of context-aware computing—mobile devices embedded in sensor networks, responding to different situations of use. This is certainly unusual within the frame of reference adopted by the "context-aware" applications discussed at the beginning of this exploration. However, the reason for this transition may be clearer in terms of the discussion as presented here. Embodied interaction encompasses more than simply physically-available interfaces. Embodiment is not about physical reality, but rather about availability for engagement. The embodied-interaction perspective is concerned with the way in which the meaningfulness of artefacts arises out of their use within systems of practice.

So, embodied interaction encompasses but extends beyond the physical, and so beyond the scope of physically-based ubiquitous computing or tangible interfaces (Ishii and Ullmer 1997). The principles underlying these interfaces—in terms of how they mediate meaningful interaction between individuals—are broader ones, and the role of context in interactive systems is broader than that conceived of by many definitions of "context-aware computing." The importance of context is not what it is but what it does in interaction—the role that it plays and the ways in which it is sustained and managed. The physical world is one in which we can see this at work, but by no means the only one. So, those who identify context-aware computing as a critical agenda for the development of interactive systems are, clearly, quite right to do so; where they may be less sure-footed is in defining context-aware computing as the responsiveness

of interactive technologies to predefined features of the times and places in which it is used.

8 Conclusions

"Context" is a slippery notion. Perhaps appropriately, it is a concept that keeps to the periphery, and slips away when one attempts to define it. The goal here has not been to define it, but rather to ask what "work" the term is doing as it is used in contemporary research in HCI. In particular, I have been concerned here with approaches to context that assume some stable separation between the "context" and "content" of activities that people might perform. This separation is at the heart of proposals to capture or represent context, which is clearly a meaningless claim if we cannot nail down at least something that is to be captured or represented. However, studies of everyday action and interaction seem to provide grounds to question this assumption.

In addition, moving away from this representational view of context lets us see that the same basic insights into how context operates are applicable beyond the traditional embedded-interactive-device model of ubiquitous computing. Indeed, if we take "ubiquitous computing" seriously, then we should be applying its ideas ubiquitously, not just in the relatively narrow areas of interaction with handheld and embedded devices.

The title of this paper is adapted from that of the Raymond Carver short story, "What we talk about when we talk about love." The genius of Carver's work is in its detailed exploration of the mundane reality of everyday life. The approach outlined here also takes the mundane details of lived experience as the basis for understanding context, not as a stable description of the world, but as the outcome of embodied practice. The examination of the unquestioned, background assumptions and practices that support everyday activity is the essence of most phenomenological analyses of the role of technology in social settings. Ethnographic accounts of technology use are becoming more familiar to researchers in HCI and ubiquitous computing, who increasingly value the "rich descriptions" and detailed accounts of encounters between people and technology. However, in this paper, I have been concerned not simply with the empirical contributions of that style of research, but with its analytic contributions—its central concern with the fact that the orderly nature of everyday conduct is an achievement of social actors, rather than something imposed upon them.

In a compelling paper, Tolmie et al. (2002) present an ethnomethodological analysis of activities in domestic settings to argue for the idea of not *ubiquitous* but *unremarkable* computing. They point out that one of the central concerns of ubiquitous computing—the invisibility of ubiquitous computing technology—is not an aspect of how the technology is designed, but rather is an aspect of how that technology is used and incorporated into practices. Just as it requires some practiced skill to notice or attend to some phenomenon, it also requires some practiced skill to be able to disattend to it, or to render it ordinary, invisible, or unremarkable in the course of its use. Like Tolmie and his colleagues, this paper advocates seeing the stable features of everyday interaction not as underlying structures to be captured and modelled, but as the outcomes of practical action. We can support the emergence and use of these structures, but we cannot separate them, analytically or technically, from the circumstances and occasions of their production.

Looking at everyday action, then, pays off in two ways. Firstly, it brings to our attention a set of problems about the ways in which context is conceived of in current design practice. Secondly, it provides us with a potential solution by furnishing us with the means to understand where our attention might instead be directed. The problem is not that context does not matter; it matters a great deal. Rather, the problem is that context is being continually renegotiated and defined in the course of action, and through this negotiation, the actions that individuals undertake can become intelligible and meaningful to each other.

Context is a central issue for HCI design and for interactive systems more broadly. The goal of the work described here is to find the right scope of the problem.

Acknowledgements The ideas expressed here have benefited from discussions with many people, including Gregory Abowd, Saul Greenberg, Tom Moran, David Redmiles, Tom Rodden, Dave Snowdon, Dag Svanaes and Jack Whalen. I am also grateful to participants at the FX PAL Workshop on Mobile Services and the HCI Consortium on Pervasive Computing, both held in 2001, for their contributions. This work has been supported in part by the National Science Foundation through awards 0133749, 0205724, and 0326105.

References

Abowd G, Mynatt E and Rodden T (2002) The human experience. IEEE Perv Comput Jan-Mar:48–57

Ark W, Selker T (1999) A look at human interaction with pervasive computers. IBM Sys J 38(4):504–507

Bowker G, Star L (1999) Sorting things out: classification and its consequences. MIT Press, Cambridge, MA

Brave S, Dahley D (1997) inTouch: a medium for haptic interpersonal communication. In: Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI'97), Atlanta, GA, 22–27 March 1997

Brumitt B, Meyers B, Krumm J, Kern A and Shafer S (2000) EasyLiving: technologies for intelligent environments. In: Proceedings of the Conference on Handheld and Ubiquitous Computing (HUC'00), Springer, Berlin Heidelberg New York

Cheverst K, Davis N, Mitchell K, Friday A and Efstreatiou C (2000) Developing a context-aware electronic tourist guide: some Issues and experiences. In: Proceedings of the ACM Conference on Human Factors in Computer Systems (CHI'00), The Hague, The Netherlands, 1–6 April 2000

Dashofy E, van der Hoek A and Taylor R (2001) A highly-extensible, XML-based architecture description language. In: Proceedings of the Working IEEE/IFIP Conference on Software Architecture (WICSA 2001), Amsterdam, The Netherlands, 28–31 August 2001

Dey A, Abowd G and Salber D (2001) A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications. Hum-Comp Interact 16(2–4):97–166

Dourish P (1995) Accounting for system behaviour: representation, reflection and resourceful action. In: Proceedings of the Third Decennial Conference on Computers in Context (CIC'95), Aarhus, Denmark, 14–18 August 1995

Dourish P (1998) Using metalevel techniques for flexibility in CSCW toolkits. ACM Trans Comp-Hum Interact 5(2):109–155

Dourish P (2001) Where the action Is: the foundations of embodied interaction. MIT Press, Cambridge, MA

Dourish P, Button G (1998) On technomethodology: foundational relationships between ethnomethodology and system design. Hum-Comp Interact 13(4):395–432

Dourish P, Lamping J and Rodden T (1999) Building bridges: customisation and mutual intelligibility in shared category management. In: Proceedings of the ACM Conference on Supporting Group Work (GROUP'99), Phoenix, AZ, 14–17 November 1999

Dourish P, Edwards K, LaMarca A, Lamping J, Petersen K, Salisbury M, Terry D and Thornton J (2000) Extending document management systems with user-specific active properties. ACM Trans Info Sys 18(2):140–170

Dourish P, Redmiles D (2002) An approach to usable security based on event monitoring and visualization. In: Proceedings of the New Security Paradigms Workshop 2002, Virginia Beach, VA, 23–26 September 2002

Fischer C (1992) America calling: a social history of the telephone to 1940. University of California Press, Berkeley, CA

Freeman E, Gelernter D (1996) Lifestreams: a storage model for personal data. SIGMOD Rec 25(1):80–86

Garfinkel H (1967) Studies in ethnomethodology. Polity Press, Cambridge, UK

Garfinkel H, Lynch M and Livingston E (1981) The work of a discovering science construed with materials from the optically discovered pulsar. Phil Soc Sci 11:131–158

Heritage J (1984) Garfinkel and ethnomethodology. Polity Press, Cambridge, UK

Hutchins EL, Hollan JD and Norman DA (1986) Direct manipulation interfaces. In: Norman & Draper (eds) User centered system design: new perspectives on human-computer interaction, Lawrence Earlbaum Associates, Hillsdale, NJ

Ishii H, Ullmer B (1997) Tangible bits: towards seamless interfaces between people, bits, and atoms. In: Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI'97), Atlanta, GA, 22–27 March 1997

Lamming M, Flynn M (1994) Forget-me-not: intimate computing in support of human memory. In: Proceedings of FRIEND21 '94, the International Symposium on Next Generation Human Interface, Magufo Jagoen, Japan, 2–4 February 1994

Ling R, Yttri B (2002) Hyper-coordination via mobile phones in Norway. In: Katz and Aakhus (eds) Perpetual contact: mobile communication, private space, public performance, Cambridge University Press, Cambridge, UK

Luff P, Hindmarsh J and Heath C (eds) (2000) Workplace studies: recovering work practice and informing system design. Cambridge University Press, Cambridge, UK

Mackay W (1990) Patterns of sharing customizable software. In: Proceedings of the ACM Conference on Computer-Supported Cooperative Work (CSCW'90), Los Angeles, CA, 7–10 October 1990

MacLean A, Carter K, Moran T and Lövstrand L (1990) User-tailorable systems: pressing the issues with buttons. In: Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI'90), Seattle, WA, 1–5 April 1990

Mark G (2002) Conventions and commitments in distributed groups. Comp Supp Coop Work 11(3–4):349–387

Marshall C, Shipman F (1993) Searching for the missing link: discovering implicit structure in spatial hypertext. In: Proceedings of the ACM Hypertext '93 Conference, Seattle, WA, 14–18 November 1993

Minneman S, Harrison S, Janssen B, Kurtenbach G, Moran T, Smith I and van Melle W (1995) A confederation of tools for capturing and accessing collaborative activity. In: Proceedings of the ACM Conference on Multimedia '95 San Francisco, CA, ACM Press, New York

Mørch A, Mehandjiev N (2000) Tailoring as collaboration: the mediating role of multiple representations and application units. Comp Supp Coop Work 9(1):75–100

Rodden T, Cheverst K, Davies K and Dix A (1998) Exploiting context in HCI design for mobile systems. In: Proceedings of the Workshop on Human Computer Interaction with Mobile Devices, Glasgow, Scotland, 21–23 May 1998

Plowman L, Rogers Y and Ramage M (1995) What are workplace studies for? In: Proceedings of the European Conference on Computer-Supported Cooperative Work (ECSCW'95), Stockholm, Sweden, Kluwer, Dordrecht, The Netherlands

Ryan N, Pascoe J and Morse D (1997) Enhanced reality fieldwork: the context-aware archeological assistant. In: Gaffney, Leusen and Exxon (eds) Computer applications in archeology, British Archaelogical Reports, Oxford, UK

Sacks H (1984) On doing 'being ordinary'. In: Atkinson and Heritage (eds) Structures of social action, Cambridge University Press, New York

Schilit B, Adams N and Want R (1994) Context-aware computing applications. In: Proceedings of the IEEE Workshop on Mobile Computing Systems and Applications, Santa Cruz, CA, IEEE Press, New York

Schilit B, Theimer M (1994) Disseminating active map information to mobile hosts. IEEE Netwk 8(5):22–32

Schön D (1983) The reflective practitioner: how professionals think in action. Basic Books, New York

Sproull L, Kiesler S (1991) Connections: new ways of working in the networked organization. MIT Press, Cambridge, MA

Strong R, Gaver W (1996) Scent and shaker: supporting simple intimacy. In: Proceedings of the ACM Conference on Computer-Supported Cooperative Work, Boston, MA, 16–20 November 1996

Suchman L (1987) Plans and situated actions: the problem of human-machine communication. Cambridge University Press, Cambridge, UK

Sudnow D (1972) Temporal parameters of interpersonal observation. In: Sudnow D (ed) Studies in social interaction, Free Press, New York

Tolmie P, Pycock J, Diggins T, MacLean A and Karsenty A (2002) Unremarkable computing. In: Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI 2002), Minneapolis, MN, ACM Press, New York

Weiser M (1991) The computer for the 21st century. Scientific American 265(3):94–104

Weiser M (1993) Some computer science issues in ubiquitous computing. Comm ACM 36(7):75–84

Wenger E (1998) Communities of practice: learning, meaning, and identity. Cambridge University Press, Cambridge, UK