The AWARE Architecture: Supporting Context-Mediated Social Awareness in Mobile Cooperation

Jakob E. Bardram and Thomas R. Hansen
Centre for Pervasive Healthcare
Department of Computer Science, University of Aarhus
Aabogade 34, DK-8200 Aarhus N., Denmark
{bardram,thomasr}@daimi.au.dk

ABSTRACT

Maintaining social awareness of the working context of fellow co-workers is crucial to successful cooperation. For mobile, non co-located workers, however, this social awareness is hard to maintain. In this paper we present the concept of *Context-Mediated Social Awareness* to denote how context-aware computing can be used to facilitate social awareness. We illustrate the concept in a case study of mobile collaboration in a hospital and present the 'AwarePhone', which is designed to support context-mediated social awareness among hospital clinicians. Based on this conceptual and empirical basis, the paper presents the AWARE architecture, which is a generic platform for supporting context-mediated social awareness.

Categories and Subject Descriptors

D.2.11 [**Software Engineering**]: Software Architecture—*Domain-specific architectures*; H.5.3 [**Information Interfaces and Presentation**]: Group and Organization Interfaces—*Computer-supported cooperative work; Asynchronous interaction; Synchronous interaction*; J.3 [**Computer Applications**]: Life and Medical Science—*Medical information systems*

General Terms

Design, Human Factors

Keywords

Context-aware computing, social awareness, mobile computing, ubiquitous computing, pervasive healthcare, AwarePhone

1. INTRODUCTION

Inherent to cooperative work is an overhead in establishing the cooperative or coordinative effort, which potentially implies interruptions and disturbances of your fellow colleagues. For example, when a nurse needs the presence of a physician in case of a severe ill patient, s/he often calls the physician on duty, thereby interrupting his or her current activity.

Sharing information about the current work context of one another is a core mechanism for initiating a proper conversation between cooperating partners. This *social awareness* helps to judge how to engage in a cooperative effort. For example, nurses spend time on maintaining a mental picture of where relevant physicians are, their current workload, and their schedule for the day. When a nurse needs to contact a physician, this picture helps her decide who to contact, when, how, and where.

Basically, social awareness relies on knowing the *work context* of a person. Hence, person A has a social awareness of person B, if A has access to the work context of B. In an operating theatre, where A and B are co-located and hence share the same work context, they can be mutually social aware [13]. If A and B are not co-located, other means of mediating information about the work context might be used. The 'online' and 'status' information in the buddy list of Instant Messaging (IM) systems is an example of computer-mediated social awareness. Studies of IM show that IM is used to provide a peripheral social awareness of fellow workers and friends [21]. These studies, however, also show that IM systems can be interruptive. IM systems rely on the discursive paradigm of interaction, and as a result IM increases the level of interruption. Presence is often negotiated with sentences like "are you there" or "do you have a moment". Hence, IM takes the opposite direction of 'social awareness' as researched in CSCW.

Furhermore, if people are mobile, the problems about negotiating presence increase. For example, when using mobile computers, a user cannot implicitly obtain the location of a person just because he is online. Mobile, pervasive, and ubiquitous computing promise computing power 'away from the desktop' which is highly relevant in much work not done at a desk as in a hospital. A core challenge is, however, to provide cooperating users in a ubiquitous computing environment with a social awareness of each another. As argued above, this basically relies on mediating an awareness about users' work context. In this paper we call this mechanism of establishing social awareness amongst colleagues through access to their working context *Context-Mediated Social Awareness*.

The purpose of this paper is twofold: First, it aims at introducing the concept of context-mediated social awareness, and second it aims at presenting a general-purpose architecture for mediating social awareness by facilitating an awareness of users' work context. This architecture applies the notion of *context-aware computing* to reveal a sense of users' context as a fundament for mediating social awareness among people. Both the conceptual discussion of context-medicated social awareness and the technical architecture are based on studies of, and design for, clinical work in hospitals.

CSCW'04, November 6–10, 2004, Chicago, Illinois, USA.

The paper is structured as follows: Section 2 provides the paper's theoretical background by defining the concept of 'context-mediated social awareness'. Section 3 and 4 provide the empirical background for the paper. Section 3 describes a field study of context-mediated social awareness in a surgical department, and section 4 describes the design, implementation, and evaluation of the 'AwarePhone' system that supports mobile social awareness. These theoretical and empirical foundations have been used in the design and implementation of the AWARE architecture described in section 5. Section 6 gives an overview of prior related systems that support social awareness. The paper is concluded in section 7.

2. AWARENESS IN CSCW

The concept of *Context-Mediated Social Awareness* aims at minimizing unwanted interruptions between mobile, distributed co-workers by enabling a social awareness through the use of context-awareness systems. In this section we present and discuss these core concepts in turn.

2.1 Interruptions

Interruptions are inherent in cooperative work. Studies of organizational and office work show that 90% of brief conversations are unplanned [31], and hence are potentially interruptive [26]. A basic asymmetry exists in the relationship between the initiator and the recipient of an unscheduled conversation; while the initiator benefits from the rapid response the recipient is interrupted in his/her current activity [21]. Studies show that only 55% of people who are interrupted continue their previous activity [22]. A common strategy for receivers of communication requests is to block incoming calls [7]. However, in many cases – especially in hospitals – this is not an option.

Seen from the initiator's point of view, a key problem is to locate and get the attention of the person with whom he/she wishes to converse [21]. About 60% of workplace phone calls fail to reach intended recipients because they are not present, or they are already talking to someone else [22, 31, 28].

These figures are not confirmed in relation to hospital work, but studies of hospitals qualitatively verify that interruptions, ad hoc conversations, and acute crisis management are core parts of working in a modern hospital [1, 5, 3, 32, 25].

2.2 Social Awareness

Ethnographic studies of co-located cooperative work show that people tacitly and unobtrusively align and integrate their activities in a seamless and highly sophisticated manner without interrupting each other. The term 'social awareness' has been used to describe this practice [27, 13]. For example studies of control rooms [14] and of operating theatres [13] reveal the subtle social mechanisms, which help people adjust their own effort according to others co-located with them. Social awareness thereby helps to minimize interruptions and disturbances when engaged in cooperative work, a phenomenon that Schmidt calls 'appropriate obtrusiveness':

> "In monitoring the work of others and in displaying aspects of their own work, actors exhibit great care and much skill in choosing an interactional modality that is obtrusive or unobtrusive to a degree and in a manner that is appropriate to the situation at hand." [27, p. 292]

Appropriate obtrusiveness can be seen as diametrically opposed to unintended interruptions. Social awareness is a mechanism for making appropriate interruptions adjusted to the nature and urgency of the current task and it is a highly active and coordinated activity. Heath and Luff use the concepts 'monitoring' and 'displaying' to highlight these aspects [13].

Monitoring is used to describe the action where an actor actively monitors the activity of other people in a setting. The act of monitoring does not disturb the person monitored. Monitoring is also characterised by being a very focused and selective process. It is only the part of the context relevant for the specific actor that is monitored. The monitoring actor combines the observed activities with an understanding of the work setting to create a kind of social awareness about the status and activities of the other participant in the setting. Thus, the monitoring actor does not always monitor the entire context, it is the actor's current activity that decides which part of the work context should be monitored [13, 27].

The opposite term of monitoring is called *displaying*. Displaying describes the implicit or explicit signals a given actor uses to show specific aspects of his or her current situation, which could be useful or relevant for the other actors in the context. Just like monitoring, displaying is a selective activity. It is the actor who selects which aspects of a given work situation to display [13, 27].

2.3 Mediated Social Awareness

Most of the research done on social awareness only discusses social awareness in settings where the people in question are co-located, or at least have visual cues about what is going on.

When non co-located people cooperate and work in a distributed manner, social awareness cannot be obtained directly but is often *mediated by social artefacts*. In hospitals, whiteboards are often used to communicate e.g. status information to all relevant personnel in a ward (e.g. [1, 32]). By combining knowledge of work settings with different cues in the environment people find hints about their co-workers' work situation. By looking at a desk, you might be able to tell if the person is at work or not, if s/he has been busy before leaving the desk, and what s/he is currently working on. Similar information can be obtained by looking in online calendars, checking meeting rooms, or looking into the operating theatre at a hospital. Hence, people use 'context cues' observed in their environment to maintain some kind of social awareness – or to put it in other words; *the working context is used to mediate social awareness*.

2.4 Context-Mediated Social Awareness

The concepts of 'context-mediated social awareness' incorporate the two terms 'monitoring' and 'displaying'. People can display their current status through the work context by leaving cues – deliberately or not. In a hospital ward, for example, a secretary can place a patient's medical record clearly visible at the edge of the desk in order to display to the doctors, that a new patient has arrived. Again people do not necessarily monitor the entire context. If a doctor does not expect any new patients s/he might not notice the record at the secretary's desk at all. The use of computers is also a mechanism for displaying and monitoring contextual information of fellow co-workers. For example, by updating (and making publicly available) an on-line calendar, or by setting the status information in an IM application.

We suggest that computers can support context-mediated social awareness by presenting remote users with context cues. Such context cues can be obtained and provided by *context-awareness* systems [10]. In this paper, we do not focus on traditional context-awareness. That is, we do not try to give the computer a sense of the user's context, but rather focus on distributing context awareness to other users of the system. Hence, some of the context information that is monitored by a context-aware system can be used as context cues to make people social aware of each other. Several projects

have shown how the context cue "where people are located" can be gathered by context-aware systems and presented to the user to support some kind of social awareness [10, 9, 30]. The weakness in many of these systems is, however, that they only support a single context cue (i.e. physical location), which could indicate several very different activities. For example, if a surgeon is situated in an operating theatre s/he might be there to operate, to get a specific instrument, to ask for advice, or something completely different. We therefore suggest that users of a context-mediated social awareness system are presented with a set of different but relevant context cues instead of just one. Furthermore, it is important to enable people to 'display' context cues about themselves.

2.5 Distributed Appropriate Obtrusiveness

Most studies of social awareness and appropriate obtrusiveness are based on studies of co-located people, or people who work in close proximity of each other, like in an operating theatre or in an office setting. It is almost impossible to maintain the same rich social awareness between people, who are distributed over a wide area, like a large hospital. However, our suggestion is to use context aware systems that supply relevant context cues to users in order to provide a context-mediated social awareness, which is useful over longer distances. This computer-based context-mediated social awareness can be used to time cooperating sessions and thereby minimize unwanted interruptions.

Furthermore, a core question in supporting appropriate obtrusiveness is to provide people with appropriate communication and cooperation mechanisms, once they need to engage in closer collaboration. If people are co-located they can use subtle mechanisms to initiate cooperation in combination with monitoring others and displaying their status. For example, by leaning forward, by looking at a person, by raising their voice, or by talking aloud. However, when people are dispersed, the possibilities available are much more limited. Basically, two types of communication can be identified. The first one is direct synchronous communication, which is supported by e.g. the telephone, an IM system, or by video conferencing. The other is asynchronous messaging systems that allow the user to leave a (prioritized) message to the recipient. The recipient can then react on it, when convenient.

By combining context-mediated social awareness with different mechanisms for initiating cooperation sessions, a person, who needs to cooperate with a remote co-worker, can choose between different suitable strategies for communication and cooperation. For example, find the co-worker and have a face to face conversation, make a phone call, send a prioritized message, choose another co-worker to cooperate with, or just wait and see if it is more appropriate to contact the person later.

We therefore suggest that it is possible to make more 'appropriate interruptions' if dispersed co-workers are supplied with some kind of context-mediated social awareness of their co-workers *as well as* different kinds of mechanisms for initiating cooperation sessions.

3. A CASE STUDY

Cooperation, coordination, interruptions, mobility, and social awareness are key characteristics of hospital work [1, 5, 3, 25, 32, 13]. To explore the notion of context-mediated social awareness we carried out 80+ hours of focused participant observation [15], held three future workshops [17], and one evaluation workshop (see section 4.1). The site chosen was the department for plastic surgery at the University Hospital of Aarhus. Nurses, surgeons, and dentists participated in the study and workshops.

Due to the specialized nature of medical work in hospitals, cooperation and hence coordination is absolutely central in achieving the flow of work. Furthermore, medical staff at hospitals are extremely mobile visiting many different buildings and floors in one day [3]. Hence, a common observation is that much time is spent looking for co-workers and trying to locate them. Furthermore, interruptions are very frequent because telephones and pagers are used extensively. A nurse, for example, is typically disturbed by doctors inquiring about a patient; by a fellow nurse who needs help; or by a patient 'pulling the string', thereby triggering the alarm.

Our studies revealed that clinicians at the department were struggling to maintain a mutual social awareness and to exercise appropriate obtrusiveness. However, because medical cooperation is distributed in time and space, this often proved difficult. Let us consider three typical work situations at the department.

3.1 Apprenticeship

Apprenticeship denotes the situation where a young and less experienced doctor has to draw on the knowledge of a more experienced doctor. The following observation illustrates this kind of situation.

> *A young doctor is treating a patient and is checking an open wound to see if it is possible to put the transplanted skin on. The wound has been bleeding a lot and is flawed close to the edges. It might on one hand be better to wait transplanting the new skin until the wound looks better, but on the other hand the transplanted skin might protect the wound and make it heal faster. The young doctor starts looking around for a more experienced doctor at the ward. Finally, he finds an experienced doctor in an office one floor up, and together they move down to the ward to look at the wound.*

This situation was very typical at the department and shows how clinicians seldom have a clue about where to find a colleague, nor what s/he does at that moment. At the workshops, it was proposed to use a computer system for providing junior doctors and nurses with an easy and fast way to contact more experienced doctors. The less experienced doctor was often engaged in e.g. treating a patient, when they needed an advice immediately. If the young doctor could be provided with some kind of social awareness about the status and whereabouts of more experienced doctors, it could help him or her in choosing the one who was nearby or did not seem to be too busy.

3.2 Cooperation between professions

Cooperation between clinical professions is extensive in a hospital, due to its high medical specialization. The following observation concerns the cooperation between a doctor and a nurse:

> *A patient wants to be transferred to a hospital closer to his home. The patient asks the nurse when he is able to leave. The nurse calls the doctor on his pager and waits. Nothing happens. Se looks at a printout of all the doctors' calendars, and the responsible doctor is apparently doing something called 'visions' today. The nurse wonders what 'visions' means and tries to find another doctor to ask if s/he knows what 'visions' is about. The other doctor has seen the doctor in question this morning and tells the nurse he will be down any minute. The nurse moves on with some other activity and later she catches the doctor and asks him to see the patient.*

When different professions need to cooperate the situation is sometimes the same as in the 'apprenticeship' case, i.e. a need for urgent help. However, in many cases a nurse, for instance, just wants to get hold of a doctor during the morning – not necessarily right away. At the workshops this situation was transformed into a vision of being able to leave 'virtual' post-it notes for the doctors. For example, a note on "Please contact me when you have a moment." Many of the nurses also spent quite some time figuring out what the doctors were actually doing at that moment. For example, in the observation above the patient was eager to go home, and he was continuously asking the nurse when the doctor would come. During the workshops it was discussed how knowledge about what the doctors current activities and hints about his plans for the day would help the nurses answer such questions.

3.3 Temporal Coordination

Most tasks in a hospital have to be done in some temporal order [1]. The patient has to arrive at a hospital before s/he can be treated; the patient has to be anaesthetized before s/he can be operated, and so on. Temporal cooperation categorizes situations where doctors or nurses are waiting for something to happen.

> *During an operation, some lymph nodes are being removed from a patient because there is a malignant cancer in the nodes. After the nodes have been removed the surgeon takes five samples from each side of the wound and sends them to the laboratory. The laboratory has to test the sample to see if the edges are free; that is, to check that the entire infected area has been removed. If the lab result is positive the wound can be closed, if the result is negative more skin has to be removed. While the lab is doing the test the doctor keeps busy doing something different, however, he is constantly checking for the lab result.*

In this case there is a need for being notified when some event has happened. In the case above it is important that the doctor react on the message right away, but in other cases it might not be as urgent. At the workshops, a need for some kind of priority mechanism was discussed. This should ensure that high prioritized messages are clearly attended to, while low prioritized messages could be delivered without interrupting the doctor's current activity.

3.4 Social Awareness and the Context

We also found that the context was used to mediate sociale awarness. In the three work situations described above the doctors and nurses lacked an awareness of other people in the work context. Because people were not co-located, it was impossible for clinicians to do displaying or monitoring of each other's activities. One way the doctors and nurses tried to address this situation was to use the context to provide awareness about the other people in the work context. The doctors' calendars were printed and placed next to a list describing in which bed to find which patients. However, patients were moved around and the doctors' schedules were changed all the time, which resulted in outdated information.

4. THE AWAREPHONE

In the future workshops, we discussed the use of computer technology to support social awareness amongst dispersed collaborating workers, thereby lowering the amount of unwanted interruptions. Based on these workshops, we identified the following requirements:

1. The system must support some kind of *context-mediated social awareness* by presenting several context cues to the users of the system.
2. The system needs to support *direct synchronous communication* that allows doctors and nurses to discuss cases in an easy way
3. The system must support the exchange of *prioritized messages* by placing a virtual post-it note on a co-worker.

Taking the need for direct communication and mobility into consideration, a high-end mobile phone was a natural choice as a platform for communication and social awareness – hence the name 'AwarePhone'.

Figure 1 shows the two main user interfaces. On the left is the *contact list* ('Kontakt' in Danish), which shows the user's list of contact persons as initials (e.g. 'JL' for 'John Larson'). Associated to each person on the list are three context cues presented: (i) 'Personal Status', which can be set from the phone, (ii) 'Activity' as displayed by an electronic calendar, and (iii) 'Location' as revealed by some automatic location detection system. These three context cues came out of the field study and was choosen at the workshops.

When selecting a contact from the list, the user can choose either to phone him or her directly, or to leave a prioritized written message. Because the AwarePhone client runs on high-end mobile phones, it can directly place calls using the embedded phone service on the device. The user can change status information by selecting his or her own initials on the list.

The right-hand side of figure 1 shows the *message list* ('Besked' in Danish), which displays incoming and active messages in a prioritized order. From this screen it is possible to read, reply and delete the incoming message.

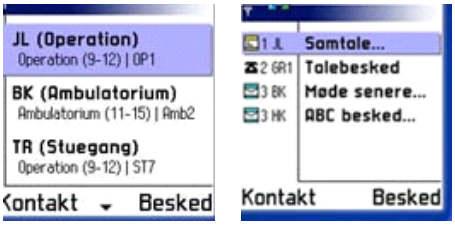

Figure 1: The User Interface of the AwarePhone. Left: The contact list with three users listed. Right: The message list with one voice message ('Talebesked' in Danish) and three text messages with a subject line.

The AwarePhone is implemented on a Nokia 7650 mobile phone running the Symbian 6.0 operating system. Using GPRS it communicates with a server, which holds a list of users and distributes the context data on location, activity, and status, as well as the messages. In the initial implementation of the server there was no concern for scalability or tailorability. This implies, for example, that the contact list for all users is the same, i.e. a list of all users. The initial AwarePhone prototype also uses a 'Wizard-of-Oz' technique for providing context information on 'Location' and 'Activity', by having a programmer manually update this information on the server.

4.1 Preliminary Evaluation

Clearly it is hard to evaluate a system, which is designed to reduce unwanted interruptions and support social awareness without putting it into use. However, to get some feedback on the design

and to evaluate the applications we held a one-day scenario workshop. Pictures from the workshop are shown in figure 2. Three doctors and three nurses participated in role-playing a set of scenarios taken from our initial observations at the department. The role-playing was followed by a semi-structured interview [23] about the idea of context-mediated social awareness. Based on this workshop we were able to draw the conclusions discussed below.

Figure 2: Images from the Evaluation Workshop.

4.1.1 Context-Mediated Social Awareness

During the role-playing sessions, some of the participants quickly started to take advantage of the social awareness information. This is illustrated by the following transcript, where a doctor needs to contact a nurse:

> **Doctor:** "But, but, I do not know where to find her"
>
> **Nurse:** "It is written there [points at the phone], I am in ward 7"
>
> **Doctor:** "Ah, okay" [looks down at the phone] [1, 4:20]

Context-mediated social awareness was especially mentioned as useful in the apprenticeship situation where a young doctor needed to contact a more experienced doctor, mainly because it allows a prioritizing or work division between the more experienced doctors. The following transcript is from the discussion about social awareness.

> **Doctor:** "It [Social Awareness] would be a clear advantage. It is a prioritizing mechanism. For instance at our ward, we have three of the 'old ones' you can contact. It would be nice if you could see who you can contact, what we are doing, if we are operating. It will always be the case, that if a doctor is in the outpatient clinic, he will be easier to disturb. It states a prioritizing order." [2, 28:06]

During the follow-up interview, we discussed the three context cues on the AwarePhone. All participants agreed on the advantages in supplying more than one context cue. The calendar information was mentioned as a good and obvious idea and not discussed further. The ability to set your own status was also mentioned as a welcome feature, as long as the different status types were general. Status text should denote activities that you plan to be doing for a while (like operating). Several of the participants made a strong point against having to change your status all the time (e.g. providing detailed status information during a surgical procedure). The last context cue was location. Most of the participants found it very useful to be able to locate other doctors and nurses, especially in large departments. The following transcript illustrates this point.

> **Doctor:** "It would be really useful to be able to find people, because when I was working in the department E, the orthopaedic department, when I was down there, it was almost impossible to find people." [2, 42:35]

However, many of the clinicians also expressed concerns about revealing location information. They anticipated a lot of organizational resistance against being tracked if location was used to monitor people. If, for example, it was possible to see the length of a coffee break or calculate the time spent together with patients. Methods for maintaining 'location-privacy' was discussed by e.g. enabling users to decide for themselves whether to reveal location information, or not to save location information over time, thereby having no history on the location of persons.

4.1.2 Methods to initiate cooperation

The ability to phone another person directly was highly appreciated – it removed the cumbersome use of pagers today, and enabled people to talk directly. Combining this with a social awareness of the recipient's current working situation helped the caller place appropriate calls.

Concerning the messaging system, the AwarePhone has a set of pre-made text messages to choose from. During the interview the adequacy of this functionality was discussed. The doctors stated that a set of standard messages would be sufficient in 90% of their cases, and the last thing they wanted was to spend time typing in messages on the phone or engaging in some sort of instance messaging on the phone. The nurses were more interested in the possibility to write messages on the phone or from a computer. In general, the nurses were interested in the ability to send short messages to the doctors, as the following two transcripts highlight.

> **Nurse:** "I would also find it nice to be able to send a message to you [the doctor], for instance related to the ward round. Then you can just send a standard message back to me." [2, 26:20]
>
> **Nurse 1:** "We have a slip of paper attached to the mirror in the office – a slip of paper where we write what we want him (the doctor) to do. And then we expect that the doctor by himself comes by and checks what we want him to do. But he does not do it! So we call his pager and run around [to locate the doctor]."
>
> **Nurse 2:** "It would be really clever if we could type in what we want the doctor to do and the doctor could then report back when he has the time." [2, 29:50]

The nurses liked the message system because much of their work was characterized by situations where they wanted the doctor to stop by, but it was often not urgent. They saw prioritized messages as a good mechanism for notifying the doctor without disturbing him. However, using the SMS-keyboard on a mobile phone was deemed problematic for most users. Hence, there was a need for providing access to the messaging functionality from other clients, like desktop computers running the electronic patient record.

5. THE AWARE ARCHITECTURE

Reflecting on this theoretical and empirical foundation, there is basis for asking what the more general technical support for social awareness would imply. The design and evaluation of the AwarePhone raised several questions:

1. How could we provide general support for context-mediated social awareness by enabling users to display and monitor a set of context cues relevant to different situations over distance in time and space?

2. How could we support multiple devices besides the AwarePhone?

3. How could we support other working environments than the surgical department? For instance other medical departments, home care nurses, or mobile information workers?

4. How can we support various context-sensing technology and data types?

5. How can we support different kinds of messaging and collaboration services?

To address these questions we have designed the AWARE architecture. The design of the AWARE architecture is based on the concept of context-mediated social awareness in mobile, distributed cooperation, and is grounded in our qualitative research of cooperative work in hospitals. The core idea in the AWARE architecture is to combine CSCW system components for providing *social awareness* among collaborating users with Ubiquitous Computing components for obtaining *context-awareness*. This two-fold strategy is also reflected in the architecture, as discussed below and shown in figure 3. Even though this architecture takes the extreme mobile collaboration in hospitals as its initial case, the goal is to make a generic design applicable in other settings characterized by mobile cooperation over distance, like mobile office workers. This section presents the design and implementation of the AWARE architecture as a framework for creating context-aware applications mediating social awareness. The AwarePhone described in section 4 has been re-implemented on top of the AWARE architecture.

The architecture of the AWARE framework is shown in figure 3 and is organized in four layers. The *Client* layer contains end-user applications using the framework as a back-end system. The *Awareness layer* contains the *Awareness Service*, which maintains an awareness of people, how they link together socially, and how to reach them. The Awareness Service is described in details in section 5.1 below. Furthermore, the Awareness layer in our current implementation contains a *Message Service*, which is a simple message broker supporting the requirement of being able to post messages to colleagues. Such simple messaging functionality is fundamental for simple communication and coordination, and hence for maintaining social awareness. Strictly speaking, the Message Service is an 'optional' part of the AWARE architecture because it could be replaced by an IM service, a SMS/MMS message service, or another more elaborate cooperation and/or workflow system. However, in the design of client applications on top of the AWARE architecture, it turned out to be valuable to have an 'embedded' messaging service in the architecture, because it enables us to embed messaging functionality directly in the client applications, like in the AwarePhone. The *Context* layer and the *Monitor and Actuator* layer are part of the context infrastructure on top of which the AWARE framework is build. This context infrastructure is further discussed in section 5.2 below.

5.1 The Awareness Service

The Awareness Service has two main responsibilities. The first responsibility is to maintain information about users subscribed to the AWARE system, about whom they want to keep a social awareness, and to handle event-based notification about changes to these users' context. This is done through close collaboration between the *Contact Manager*, the *Entity Listener*, and the *Message Listener*. The second responsibility of the Awareness Service is to handle connections to clients, using different protocols, respond to their requests, and to know how to notify them about relevant events. This is done through the *Awareness Gateway* component.

The Contact Manager keeps track of the contacts for each user, i.e. about whom this user wants to keep a social awareness about. This is a simple list of other users in the AWARE system, much like the list of contacts in an IM system.

An important quality of the AWARE architecture is its event-based infrastructure. Hence, when changes to a user's context take place, relevant clients are notified. For example, if person A has person B in his list of contacts, then A is notified if person B moves from one location to another, or other context information is changed. Similarly, if person X sends a message to person Y, and person Y reads this message, then this information is propagated to person X's device. Applications running on the device can now treat these events appropriately, i.e. updating the user-interface or notifying users. This event-based quality of the architecture is accomplished by the *Entity Listener*, which is listening to changes in context information in the Context Service, and by the *Message Listener*, which is listening for new messages or status changes to messages.

In order to support different kinds of client devices, the Awareness Gateway is able to transform protocol-specific requests to the Awareness Service and translate this into using the internal AWARE API for responding to clients. This API is the collection of the APIs from the Context Service, the Awareness Service, and the Message Service, respectively. Examples of requests are:

- `getAllContacts()` returns all contacts for a user using the Awareness Service.
- `sendMessage()` sends a message using the Message Service.
- `setContextItem()` changes some context information in the Context Service, e.g. status information for a user.

An important responsibility of the gateway is to know how to contact a user, by knowing his current device (e.g. a mobile phone), its communication protocol, and relevant technical information on physical addresses and communication port numbers. This is used for event-notification. The Awareness Gateway is designed according to the 'Strategy' design pattern [12] by having *Protocol Converters* as plug-ins to the gateway. These protocol converters map between one protocol (e.g. HTTP) and the API used by the services running in the AWARE Architecure.

Currently, we have built converters for HTTP, Java RMI, and special-purpose PHO protocol for communicating with the AwarePhones. The PHO protocol allows clients to communicate with the server using a TCP/IP socket interface. This simple protocol was designed and used instead of e.g. XML over HTTP (SOAP) or any other kind of RPC / RMI interface because of the limited bandwidth and processing power on the small devices we want to use. For example, the Nokia 7650 mobile phone has limited CPU and memory and the communication to the gateway is TCP/IP over GRPS, having a typical bandwidth on 30 kbits/s upstream and 10 kits/s downstream. Furthermore, users pay pr. kb downloaded. Hence, when

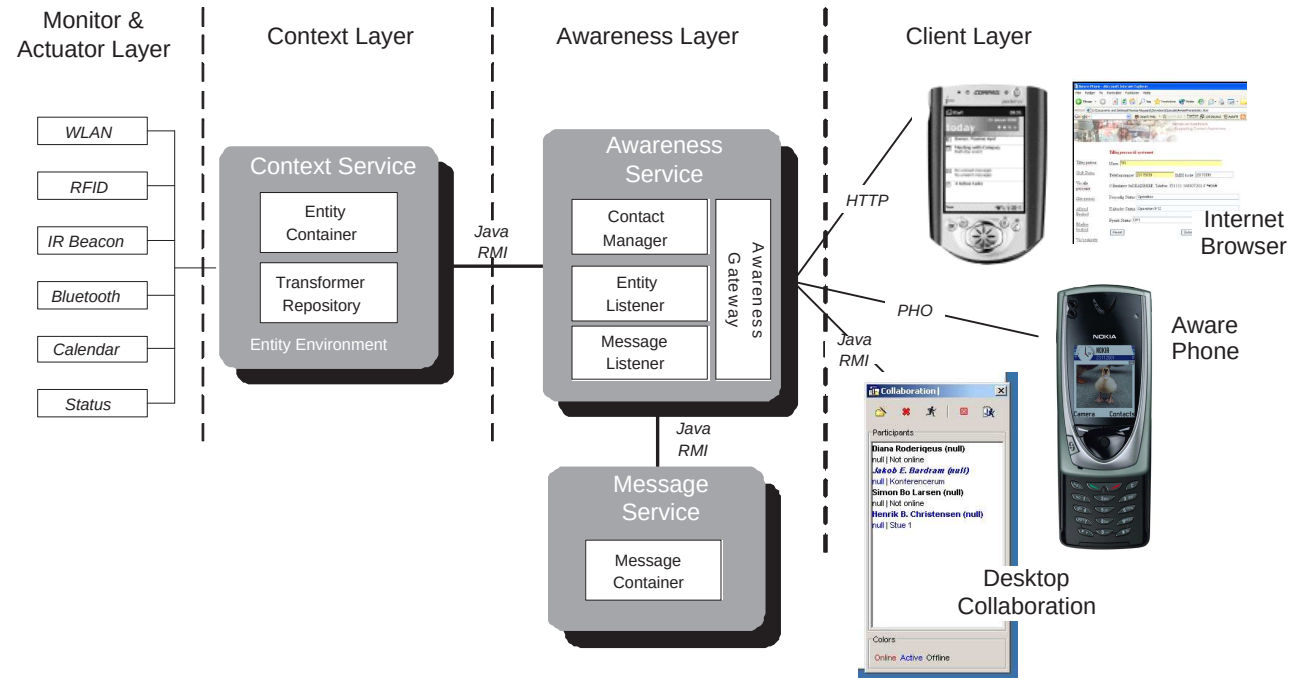

Figure 3: System architecture for the AWARE framework. The architecture is divided into four layers: the Client layer, the Awareness layer, the Context layer, and the Monitor and Actuator layer.

creating an event-based notification infrastructure, the notification protocols must be as compact as possible. As an example, the message for adding or changing contact information for a contact listed on the phone is illustrated below:

```
CONIF:86868686#thomas@xxx.yyy
      #Busy
      #Meeting 8-10
      #Conference room 2388.115§
```

Event-notification is supported in the PHO and RMI protocols. HTTP is a stateless request-response protocol and hence does not support this event-notification.

5.2 The Context Infrastructure

The context infrastructure is responsible for monitoring context cues in the users' environment, including displaying cues from users, and for storing, managing, and distributing these cues in the awareness service. The context infrastructure builds on top of the Java Context-Awareness Framework (JCAF) [2]. In order to support various context sensing technologies and data types, the context infrastructure is divided into a 'Context layer', responsible for managing, transforming, and distributing context information, and a 'Monitor and Actuator layer', responsible for context acquisition using various sensing and actuator technologies. The left-hand side of figure 3 illustrates the main components of this context-awareness sub-system: the Context Service; and the Context Monitors and Actuators.

A *Context Service* is a long-lived process running on e.g. servers in a distributed infrastructure. Context information about the real world is modelled by the concept of an *Entity*, which has some associated *Context* information. Examples of entities are persons, places, things, patients, beds, pill containers, etc. Examples of context information are location, noise level, a person's activity, or people nearby. Entities and their context information are managed by the context service's *Entity Container*. An entity is represented as a small Java program that runs within the Context Service and responds to changes in its context. The life cycle of an entity is controlled by the container in which the entity has been added. The entity container handles subscribers to context events and notifies relevant clients on changes to entities. An entity, its context and its life cycle is further discussed below.

The Entity components in a Context Service work together and with other components to accomplish their tasks. Hence they must have ways to access each other and to access shared resources, like database connections or RMI stubs to other processes. This is accomplished through the *Entity Environment*, to which all Entities have a handle when executing. Besides access to general resources like initialization parameters and logging facilities, the Entity Environment provides methods for accessing *Key-Value Attributes* and *Context Transformers*. Context transformers are small application-specific Java programs that a developer can write and add to the *Transformer Repository*. The Transformer Repository can be queried for appropriate transformers on runtime.

Context Clients can access entities and their context information in two ways. Either following a request-response schema, requesting entities and their context data, or by subscribing as an *Entity Listener*, listening for changes to specific entities. JCAF also supports *type-based* subscriptions of entity listeners, allowing a client to subscribe to changes to all entities of a specific type.

There are two special kinds of context clients: the *Context Monitor* and the *Context Actuator*. A Context Monitor is a hardware and/or context data specific process, which registers changes in the physical or digital environment. The monitor adapts this context information according to the data model in the Context Service. This 'Context Monitor' abstraction is similar to the Context Toolkit [10]

where the monitor component in the JCAF framework corresponds to a combined widget and interpreter in the Context Toolkit. Examples of context monitors are location monitors based on GPS receivers, or WLAN monitors which try to locate WLAN-based equipment. Other monitors might gather information about temperature, planned activities in users' personal calendars, or status information in an Instant Messaging system. A context actuator is a client designed to work together with hardware actuators, like loud speakers, displays, or hydralic lifting columns for beds or tables. A context actuator could also be designed to change status in an IM system from the AwarePhone built on top of the AWARE architecture.

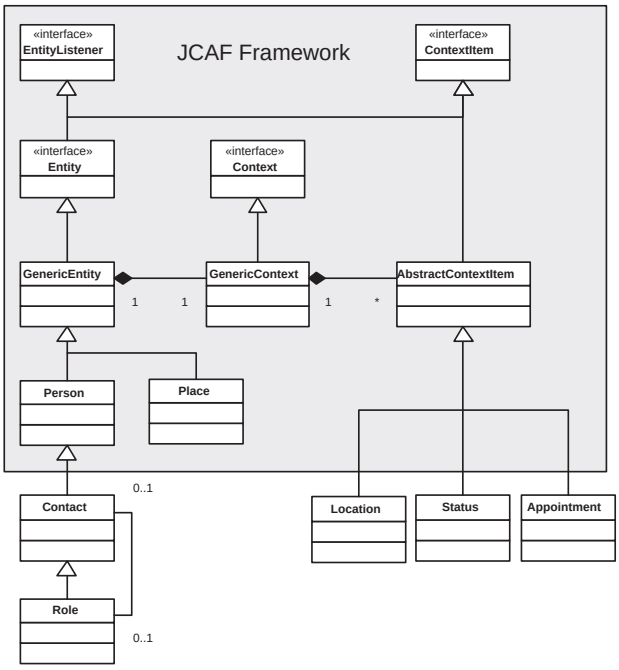

Figure 4: The UML model of AWARE specific specialization of the JCAF framework.

The basic modelling concept in the JCAF API is the `Entity`, which has a `Context` with a set of `ContextItem`s. An entity, a context, and a context item are all Java interfaces, which developers of context-aware applications must implement. Figure 4 illustrates how these JCAF primitives are used in the AWARE framework. The `Person` and `Place` are default entities in JCAF and represent a person, like a user of the AwarePhone, and a physical place, like an office, respectively. Examples of context are a Hospital Context and an Office Context, each knowing specific aspects about a hospital and an office, respectively. Examples of context items are physical location, appointment as revealed by a user's calendar, and the patient in a hospital bed. Context items are added to an entity's context typically by context monitors or other clients. Hence, the context item `Location`, which models a physical location, can be added to an entity, thereby registering the location of this entity. A subtle, but rather important aspect of entities is that they themselves are context items (`Entity` extends `ContextItem` in figure 4). Hence, a person (an entity) can be added to the context of a room (another entity) thereby indicating that this person is in this room's context. If the person's context is changed then the room would be notified, and hence entity listeners to the room would also be notified.

The Awareness Service uses the context infrastructure by connecting to a Context Service and registering itself as a listener to entities relevant for the AWARE framework. These are entities of the type `Contact` and its sub-type `Role` in the context service. The Awareness Service then receives notifications via the entity listener when changes to these contacts take place. For example, the Awareness Service is notified when a person changes location, changes his personal status in a client, or he starts a meeting according to his calendar. To achieve this, the AWARE framework has created and registered three types of Context Monitors in the context service: a Status monitor, monitoring status change in an IM system; a Calendar Monitor, monitoring an user's online calendar; and some Location Monitors, monitoring location of users. These three types of monitors add the `Status`, `Location`, and `Appointment` context items to a contact's context.

5.3 Implementation Status

The current version of the AWARE architecture is implemented in Java J2SE 1.4. We have created three clients which use the architecture; a modified version of the AwarePhone described in section 4, a HTML client running in a browser on a PDA, and a Java-based client running on a desktop PC. These three clients are illustrated in figure 3. We have created a version to be used in a hospital environment (primarily based on our experience from the work with the surgical department and the AwarePhone) as well as a version to be used in our own office environment at the university. The AwarePhone now runs on Symbian version 7.0s and can hence be used on the newer version 6600 Nokia telephone, which is extensively used at our university.

The activity sensor monitors use appointments in the MeetingMaker online calendar [18], which is used at our university. Currently we have implemented three types of location sensors to the AWARE platform based on Bluetooth beacons, Infrared (IR) beacons, and cell-based location based on WLAN base stations.

The IR and Bluetooth location mechanisms work in a similar way. Beacons emitting a location URL are placed in various places (typically rooms). This location information is captured by the mobile phone's IR or Bluetooth port respectively. Initially, we used infrared beacons in each room to locate the device. The advantage with IR is that it stays within the current room. However, IR also requires line-of-sight between the IrDA port on the phone and the beacon. Often the phone was in the pocket of a user, and hence the IR location did not work. The advantage of Bluetooth is that it does not require line-of-sight. The disadvantage of Bluetooth is that it is not limited to the current room and small beacons typically have a range at approximately 10 meters. Currently, we are experimenting with the BlipNet infrastructure from Blip Systems [4]. This system has an adjustable range from 0.60 to 10 meters and can be used for more precise location of bluetooth devices.

In the IR and Bluetooth location model, a mobile phone is able to sense its own location. In contrast to this model, the WLAN location sensor is a central component that queries WLAN base-stations about which WLAN devices are within its range. This gives a coarse grained, cellular location information by indicating a non-specific area around the WLAN base-station.

As discussed in section 4.1, the users expressed concerns about location privacy. In our current implementation of the AWARE architecture context has no history, i.e. no context information is saved over time. For the time being, this seems a fairly appropriate solution. But there is a need for making more elaborate mechanisms for protecting user privacy concerning context information.

6. RELATED WORK

In this section we will discuss related work with regard to computer support for social awareness among distributed cooperating users. Related work on the concept of 'context-mediated social awareness' has already been addressed in section 2.

In a study similar to ours, Muñoz et al. [20] address the need for context-aware communication in hospitals. They reach some of the same conclusions concerning the importance of providing contextual information like location, role, and status to medical personnel in hospitals. Based on this, they extend the IM paradigm to include context-awareness, thereby enabling clinicians to send a message by specifying context instead of specifying a recipient. Furthermore, clinicians can locate each other on a map of the hospital ward. Their focus is slightly different than ours, focusing on communication based on context-aware messaging. Our focus is on using contextual information to mediate social awareness, thereby enabling people to communicate as they please, including sending messages, calling them using the phone, or go to them and engage in face-to-face conversation. Furthermore, there are some technological differences. The context-aware IM system uses PDAs connected via WLAN, which is also used for location estimation using triangulation. The AwarePhones are connected via GPRS, using Bluetooth for location, and the PDA HTTP/HTML clients are using WLAN for communication and for cell-based location[1]. During our evaluation sessions, we discussed the use of a map on the phone. The clinicians, however, deemed that this was of little use, since they know the layout of the hospital very well and providing a text-based location was actually more efficient than a map. The only use was maybe for new clinicians starting internship at the hospital, or for patient and relatives. However, few clinicians would like to be located by patients and relatives.

Michell et al. have built a 'follow-me' video application for hospitals on top of the QoS Dream framework [19]. Their focus, however, is not on providing a social awareness within a hospital but to locate the receiver of a conference call and to enable this conference to follow the users while moving.

Other research prototypes have explored how to extend the IM paradigm to also include cues on e.g. location, online calendar information, and other context information in the contact list – both in desktop computers and in mobile PDAs [6, 11, 20, 29]. For example, the 'Awarenex' system extends the 'ConNexus' IM system to mobile devices (the Palm) and supports a mutual awareness in the contact list by listing people, their location, and their current online schedules [29]. Similarly, by making part of the physical location of tourists publicly available, the GUIDE system aims at enabling a form of social awareness amongst city visitors [9].

In terms of software architecture, these systems are built from the ground up with specific applications in mind. Thus, the opportunity to reuse software components in other applications is limited. We would argue, that the main contribution of our work is the more general-purpose AWARE platform lying beneath e.g. the AwarePhone client. This AWARE platform combines support for context-awareness with support for social awareness on a software architectural level. Hence, in the AwarePhone, the context items of location, status, and activity were used, but a more rich picture of context might be added by creating or reusing appropriate context monitors and items. For example, we could extend the list on the AwarePhone to include context information about places, like operating theatres, wards, etc. Then users would have access to virtually 'look into' these places to get an awareness about the working context in this place. Such an application can be built using the AWARE platform. In addition, new monitors and items may be added on runtime to the Context Service in the AWARE platform and are available to client applications – like the AwarePhone – immediately. Furthermore, context information already available in the context service infrastructure can be re-used. This extensibility of the AWARE platform also applies for the Awareness Service. Plug-ins for new types of clients can be made and added to the Awareness Gateway. As an illustration, the AWARE platform enabled us to create a PDA version similar to the 'context-aware IM system' which were integrated with other clients, like the AwarePhone.

Finally, the messaging system in the AWARE platform is not a chat system like IM. The messages in the AWARE platform are more like post-it notes left for relevant people (or roles) in right places, like the Stick-e Document System [8]. Hence, our approach is the opposite of IM, which relies on the discursive paradigm of interaction, and as result might increase the level of interruption. By contrast, and in direct continuation of the 'awareness' research in CSCW, our work explores ways to support "appropriate obtrusiveness" by indirect means (location etc.), as opposed to providing "social awareness" by constant interruptions.

[1]We deliberately did not try to make WLAN triangulation. Firstly, because it is already extensively researched and commercial systems are now available. Secondly because WLAN triangulation is very difficult in hospitals with very non-uniform layouts and materials causing a lot of RF reflection.

7. CONCLUSION AND FUTURE WORK

Context-Mediated Social Awareness is a new term introduced in this paper. The term does not introduce a new work practices, but emphasises how the context can be used to provide social awareness between co-workers working in different locations. We have looked into context-mediated awareness in a hospital setting, and shown how the term can be used to describe different work practices. The 'AwarePhone' is an example of a computer application to support context-mediated social awareness.

Based on the work done specific to hospitals, we have designed and implemented the AWARE architecture; a general-purpose software architecture for mediating 'social awareness' in different settings. The architecture presents new ways of supporting awareness by using a context-awareness infrastructure as the basis for mediating social awareness.

We have built several clients on top of the architecture and we are currently planning a pilot deployment of the AWARE architecture and some of the AWARE clients in a hospital. The pilot phase is planned to run for three months, which should give us the opportunity to make extensive evaluations of the architecture and its effect on minimizing inappropriate obtrusiveness in hospitals. We are also looking into how to deploy the architecture in an office environment and how the architecture can be integrated with the existing systems here.

Acknowledgments

This research is funded by ISIS Katrinebjerg. Claus Bossen, Jane Clemensen, and Simon Bo Larsen participated in the evaluation of the system. Mads Ingstrup implemented the MeetingMaker and WLAN context monitors.

8. REFERENCES

[1] J. E. Bardram. Temporal Coordination: On Time and Coordination of Collaborative Activities at a Surgical Department. *Computer Supported Cooperative Work. An International Journal*, 9(2):157–187, 2000.

[2] J. E. Bardram. The Java Context Awareness Framework (JCAF) – A Service Infrastructure and Programming Framework for Context-Aware Applications. Technical Report CfPC Technical Report 2004–PB–61, Centre for Pervasive Computing, Aarhus, Denmark, 2003. Available from http://www.pervasive.dk/publications.

[3] J. E. Bardram and C. Bossen. Moving to get aHead: Local Mobility and Collaborative Work. In K. Kuutti, E. H. Karsten, G. Fitzpatrick, P. Dourish, and K. Schmidt, editors, *Proceedings of the Eighth European Conference on Computer Supported Cooperative Work*, pages 355–374, Helsinki, Finland, Sept. 2003. Kluwer Academic Publishers.

[4] Blip Systems. http://www.blipsystems.com/.

[5] C. Bossen. The Parameters of Common Information Spaces: The Heterogeneity of Cooperative Work at a Hospital Ward. In *Proceedings of the 2002 ACM Conference on Computer Supported Cooperative Work*, pages 176–185. ACM Press, 2002.

[6] E. Bradner, W. A. Kellogg, and T. Erickson. The adoption and use of 'babble': A field study of chat in the workplace. In S. Bødker, M. Kyng, and K. Schmidt, editors, *Proceedings of the Sixth European Conference on Computer Supported Cooperative Work*, pages 139–158, Copenhagen, Denmark, Sept. 1999. Kluwer Academic Publishers.

[7] B. Brown and M. Perry. Why don't telephones have off switches? Understanding the use of everyday technologies; A research note. *Interacting with computers*, 12(6):623–634, 2000.

[8] P. J. Brown. The Stick-e Document: a Framework for Creating Context-aware Applications. In *Proceedings of EP'96, Palo Alto*, pages 259–272, January 1996.

[9] K. Cheverst, K. Mitchell, N. Davies, and G. Smith. Exploiting Context to Support Social Awareness and Social Navigation. *SIGGROUP Bull.*, 21(3):43–48, 2000.

[10] A. Dey, G. D. Abowd, and D. Salber. A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications. *Human-Computer Interaction*, 16:97–166, 2001.

[11] H. Fagrell, K. Forsberg, and J. Sanneblad. Fieldwise: a mobile knowledge management architecture. In Kellogg and Whittaker [16], pages 211–220.

[12] E. Gamma, R. Helm, R. Johnson, and J. Vlissides. *Design Patterns: Elements of Reuseable Object-Oriented Software*. Addison-Wesley, 1994.

[13] C. Heath, D. vom Lehn, J. Hindmarsh, M. Svensson, Sanchez, and P. Luff. Configuring Awareness. *Computer Supported Cooperative Work. An International Journal*, 11(3-4):317–347, 2002.

[14] C. C. Heath and P. Luff. Collaboration and control: Crisis management and multimedia technology in London Underground control rooms. *Computer Supported Cooperative Work. An international Journal*, 1(1-2):69–94, 1992.

[15] B. Jordan. Ethnographic Workplace Studies and CSCW. In D. Shapiro, M. Tauber, and R. Traunmüller, editors, *The Design of Computer Supported Cooperative Work and Groupware Systems*. Elsevier, 1996.

[16] W. Kellogg and S. Whittaker, editors. *Proceedings of the 2000 ACM Conference on Computer Supported Cooperative Work*. ACM Press, 2000.

[17] F. Kensing and K. Halskov Madsen. Generating Visions: Future Workshops and Metaphorical Design. In J. Greenbaum and M. Kyng, editors, *Design at Work: Cooperative Design of Computer Systems.*, pages 155–168. Lawrence Erlbaum Associates, Hillsdale, NJ, 1991.

[18] MeetingMaker Homepage. http://www.meetingmaker.com.

[19] S. Mitchell, M. D. Spiteri, J. Bates, and G. Coulouris. Context-Aware Multimedia Computing in the Intelligent Hospital. In *Proceedings of the 9th ACM SIGOPS European Workshop*, pages 13–18. ACM Press, 2000.

[20] M. A. Muñoz, M. Rodríguez, J. F. Center, A. I. Martinez-Garcia, and V. M. González. Context-Aware Mobile Communication in Hospitals. *IEEE Computer*, 36(9):38–46, 2003.

[21] B. A. Nardi, S. Whittaker, and E. Bradner. Interaction and Outeraction: Instant Messaging in Action. In Kellogg and Whittaker [16], pages 79–88.

[22] B. O'Conaill and D. Frohlich. Timespace in the Workplace: Dealing with Interruptions. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 262–263. ACM Press, 1995.

[23] M. Q. Patton. *Qualitative Evaluation and Research Methods*. Sage Publications, London, second edition, 1990.

[24] W. Prinz, M. Jarke, Y. Rogers, K. Schmidt, and V. Wulf, editors. *Proceedings of the Seventh European Conference on Computer Supported Cooperative Work*, Bonn, Germany, Sept. 2001. Kluwer Academic Publishers.

[25] M. Reddy, P. Dourish, and W. Pratt. Coordinating heterogeneous work: Information and representation in medical care. In Prinz et al. [24], pages 239–258.

[26] M. Rouncefield, S. Viller, J. Hughes, and T. Rodden. Working with constant Interruption: CSCW and the Small Office. *The Information Society*, 11(4):173–188, 1995.

[27] K. Schmidt. The Problem with 'Awareness': Introductory Remarks on 'Awareness in CSCW'. *Computer Supported Cooperative Work. An International Journal*, 11(3-4):285–298, 2002.

[28] J. C. Tang and M. Rua. Montage: Providing Teleproximity for Distributed Groups. In J. B. Smith, F. D. Smith, and T. W. Malone, editors, *Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work*, pages 37–43. ACM Press, 1994.

[29] J. C. Tang, N. Yankelovich, J. Begole, M. Van Kleek, F. Li, and J. Bhalodia. Connexus to awarenex: extending awareness to mobile users. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 221–228. ACM Press, 2001.

[30] K. Tollmar, O. Sandor, and A. Schömer. Supporting Social Awareness Work Design and Experience. In G. Olson, J. Olson, and M. S. Ackerman, editors, *Proceedings of the 1996 ACM Conference on Computer Supported Cooperative Work*, pages 298–307. ACM Press, 1996.

[31] S. Whittaker, D. Frohlich, and O. Daly-Jones. Informal workplace communication: what is it like and how might we support it? In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 208 –. ACM Press, 1994.

[32] Y. Xiao, C. Lasome, J. Moss, C. Mackenzie, and S. Faraj. Cognitive properties of a whiteboard: A case study in a trauma centre. In Prinz et al. [24], pages 259–278.