Attentive Groupware Systems: A Framework and a Prototype Tool

António Ferreira
Pedro Antunes

DI–FCUL TR–07–31

November 2007

Departamento de Informática
Faculdade de Ciências da Universidade de Lisboa
Campo Grande, 1749–016 Lisboa
Portugal

Attentive Groupware Systems: A Framework and a Prototype Tool

António Ferreira Pedro Antunes
Department of Informatics, University of Lisbon, Portugal

November 2007

Abstract

We address the development of software devices that can be installed on groupware systems to help people stay attentive and be more productive during computer-mediated collaboration. We introduce a framework that derives attentive devices from groupware mediation modes to take into account attentional phenomena, and demonstrate the framework's role in an attentive electronic brainstorming tool. Finally, we describe a laboratory experiment to evaluate the impact on group performance caused by the use of the attentive devices, and present preliminary results that encourage further evaluation on the opportunity seeker, which minimises interruptions while the user is typing an idea.

1 Motivation

One of the goals of groupware is to improve the sense of proximity within groups, whether by enabling geographically distant people to work together, or by supporting ongoing activities at different times. To this end, researchers have been proposing several mechanisms to enhance group awareness by providing *ever greater* information about the presence and actions performed by people, e.g., radar views, multi-user scrollbars, and telepointers [1]. The main argument is that communication channels mediated by computers are relatively poor when compared with more natural settings, such as face-to-face meetings [2]. However, a problem with this trend is that it fails to recognise that sometimes *more is less* due to the limitations in the human attentive capacity, especially as we become surrounded with computers and, not necessarily useful, information [3].

During the late 1990s several researchers from the Human-Computer Interaction (HCI) area have became interested in Attentive User Interfaces (AUI), and since then this topic is gaining momentum, as evidenced by a special issue in Communications of the ACM [3], and another in Computers in Human Behavior [4]. The prime motivation for AUI is the recognition that as the needs for information and communication rise so do the costs of not paying attention and being interrupted [3]. So, instead of assuming the user is always focused on the entire computer display, AUI negotiate the users' attention by establishing priorities for presenting information.

Most research in AUI is directed towards single-user activity, the main assumption being that individual performance degrades with the number of simultaneous requests for attention. Therefore, researchers are enhancing input/output devices so that the user remains focused on a primary task without getting too much distracted by a secondary, typically unrelated, task, e.g., by using eye-trackers [5], statistical models of human interruptibility [6, 7], displays that show information at various levels of detail [8], and other devices.

Regarding multi-user activity, the research is mainly situated in video conferencing [9, 10], making the study of attention in computer-mediated collaborative contexts a largely unexplored area. Moreover, the convergence of groupware and AUI poses new challenges to researchers due to differences in individual and group work, namely:

- People working in a group attend to more interruptions than in individual work because they have to manage more information flows;
- Instead of doing a single, extensive task, group members usually execute a series of intertwined tasks;
- In group work the primary and secondary tasks are typically related and may both contribute to the shared goal.

Furthermore, the current emphasis in groupware research that applies AUI concepts is still on evaluating the enhanced input/output devices themselves, in contrast with determining the outcomes of using these devices in work settings.

Given this situation, in which the validity of existing results may be reduced if directly applied to groupware systems, we propose a framework for mediating attention in groups and put forward the possibility of using specialised, software-based, groupware devices to account for attentional phenomena. We intend to reuse some knowledge gathered over the decades by psychologists concerning the goals and limitations of human attention, to design and develop *attentive devices*. These devices manipulate the information flows supported by the groupware system, and since remote collaboration is entirely mediated by computers, its use might mean that the group as a whole would be more attentive, and possibly more productive.

We explain the framework for attentive groupware systems in Sect. 3, and in Sect. 4 we describe its application to an electronic brainstorming tool and provide examples of how the attentive devices work in this context. In Sect. 5 we describe a laboratory experiment that we set up to evaluate the impact on group performance caused by the use of the attentive devices, and present some preliminary results. In Sect. 6 we conclude the paper with a discussion regarding our approach and with prospects for future work.

2 Related work

The study of the mediating role of computers on attention within groups is largely an unexplored research area, with the exception of video conferencing. Currently, the major part of the literature on AUI is focused upon single-user software, or singleware. We will refer to both multi and single-user contexts to provide a more comprehensive picture of the systems and devices that address human attention in HCI.

2.1 Attentive groupware systems

GAZE is a groupware system developed to facilitate the detection of who is talking to whom in remote meetings [11]. It works by displaying photographs of users on the computer display, which can be rotated by intervention of eye-trackers placed in front of each user. For example, the photos might be rotated towards a photo of the user who is speaking. In this way, turn taking may be more natural and require less interruptions to determine who will speak next.

The GAZE-2 system was developed to support multiple conversations at the same time [9]. Each user has three video cameras that capture the user's face at slightly different angles; then, an automatic camera director chooses one video stream taking into account the direction where the user is looking at. As in the original GAZE, the representation of each user is rotated to reflect the focus of attention. Moreover, as the angle of rotation increases, the quality of the video stream is purposefully reduced to save network bandwidth. According to the authors, this technique is effective because it is based upon our own natural limitations concerning peripheral vision.

Another feature of GAZE-2 is the automatic filtering of voices when multiple conversations are being held at the same time. Depending upon the user in focus, so is the respective audio stream amplified, and the other streams attenuated (but not eliminated). If the focus of interest suddenly changes, as sensed by the eye-tracker, the audio is again adjusted.

Recent work with groupware systems further explore the ideas in GAZE-2. For instance, eyeView supports large meetings by manipulating the size of the video windows and the voice volumes of each user on the group as a function of the current focus of attention [10].

2.2 Attentive devices on singleware systems

In contrast with groupware systems, several input/output devices have been tested on attentive interfaces for single-user applications, such as: a) sensors that detect the user's focus of attention based upon eye-gaze and body orientation [5, 12, 13]; b) physiological sensors that assess the user's mental workload by measuring heart rate variability, pupil dilatation, and eye-blink activity [14, 15, 16]; c) statistical models that determine adequate moments to interrupt and communicate with the user [6, 7]; and d) displays that present information at various levels of detail, depending on the user's focus of attention [8].

Regarding the use of eye-trackers to support human attention, applications include enlarging the graphical window which the user is currently focused on, controlling a robotic directional microphone coupled to a video camera, to overhear a particular conversation taking place in a remote room, and detecting eye contact to automatically choose which electronic appliance should obey to voice commands [5]. Other applications use eye-gaze to position a cursor on the screen with minimal hand intervention [12] and to help a user read a book written in a foreign language [13].

Body orientation sensors have been tested in office environments to regulate the transparency of cubicle walls (opaque when the user does not wish to attend requests from others) and to control noise cancellation in headphones [5].

Concerning physiological sensors, these have been used to assess mental workload, which, in turn, is considered a surrogate of the user's attentional state. One study suggests using heart rate variability and electroencephalogram analysis to distinguish between at rest, moving, thinking, and busy states, and describes an automated regulator of notifications for mobile phones [15]. Heart rate variability had already been used in the 1990s to assess conditions of excessive mental effort [16]. More recently, pupil dilatation has been used together with hierarchical task models to predict opportune moments to interrupt the user [14].

Another approach to detect the best time to interrupt the user is to apply statistical models that permanently estimate and balance the value of information with the cost of interrupting, based upon a stream of clues, such as, appointments on the personal calendar, past activities and work patterns, ambient noise, or body posture [6]. Statistical models have also been used to select the best predictors of interruptibility from a myriad of software sensors embedded in an integrated development environment [7].

Finally, we also refer to a display that saves rendering resources by adjusting the level of detail in selected areas of the screen as a function of the user's current visual focus of attention [8]. This is similar to the GAZE-2 approach in that in both cases it is accounted that our peripheral vision has low sensitivity to details.

2.3 Discussion

Research in AUI, its applications and devices, is progressing in many directions. However, we argue that most studies are directed towards evaluating the devices *per se*, in contrast with determining the outcomes of using these devices in work settings. For example, to the best of our knowledge, the GAZE-2 system was evaluated through a user questionnaire that measured the self-perception of eye-contact and distraction, as well as changes in colour and brightness during camera shifts [9], but no attempt was made to determine if group work benefited. The same situation seems to occur with eyeView [10], which is still in an early stage of development.

Some studies do address the evaluation of AUI from the perspective of task execution, but are restricted to single-user activity. One study measured the effects of interruption on completion time, error rate, annoyance, and anxiety, and suggests that AUI should defer the presentation of peripheral information until task boundaries are reached [17]. In another study, the effectiveness and efficiency of users were evaluated as they performed two types of tasks under the exposure of four methods for coordinating interruption, and it recommends that AUI should let users manually negotiate their own state of interruptibility, except when response time for handling interruptions is critical [18].

We note, however, that there are numerous differences in individual and group work that might reduce the external validity of current results. This opens an opportunity for doing research in groupware systems and AUI.

3 Framework for attentive groupware systems

The purpose of this framework is to conceptualise attentive groupware systems. Its underlying assumption is that group performance may improve by incorporating human attentional phenomena in attentive devices that adjust the

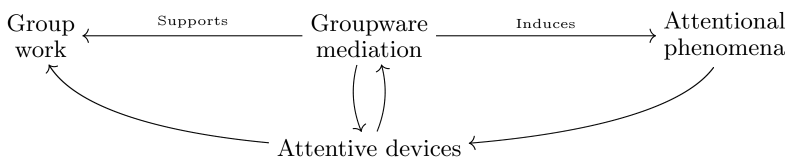


Figure 1: Conceptual framework for attentive groupware systems

groupware mediation, e.g., the information in circulation, as people carry out the collaborative tasks (see Fig. 1). We note that evaluation of group work is outside of the scope of the framework but is strongly tied to its utility.

In the next sections we describe the concepts of the framework, based upon extensions of our previous work on this subject [19].

3.1 Groupware mediation

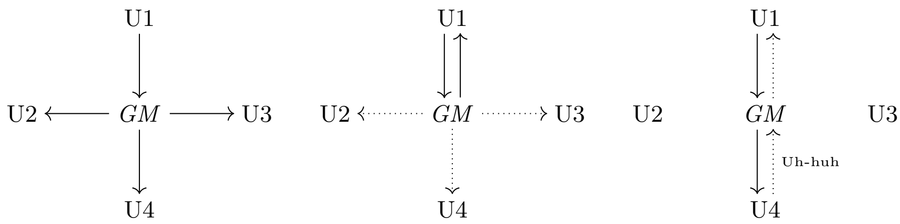
Conceptually, the groupware system is at the centre of the group. It is a mediator that deals with all sorts of information that comes and goes between and among the users. Moreover, we assume that users are restricted to using groupware to collaborate, such as when they are geographically distributed, since this is increasingly common in organisational and other social settings and also because this is a challenging scenario for managing attention in groups.

Groupware mediation should support the notions of interdependence and mutual awareness (e.g., for group planning and situation assessment, respectively), which generate information that requires human attention in order for the group to make progress. Our strategy to characterise groupware mediation is to change the perspective over the information flows that we investigated in a previous study about shared workspaces [20]: instead of looking into the ways remote people collaborate, we analyse the corresponding mediation modes.

In the *explicit communication mode* the groupware receives information produced by a user and forwards it to one or more users, based upon an explicit request by the sender [2]. This may happen when a user expresses a request for an object to the user who is holding it; another example is when an instructor provides online guidance to students for collaborative problem solving (illustrated in Fig. 2a). This mode can be supported by a groupware interface capable of multiplexing information from input devices to several output devices, e.g., a user typing on a keyboard and the other users seeing the text on their displays.

In the *feedthrough mode* the groupware automatically reports actions executed by one user to several users [21]. This mode is essential because it provides group awareness and enables users to construct meaningful contexts for collaboration. For example, a graphical shared workspace may provide its users with information about the menu selections for each user who is manipulating objects. The groupware interface can support the feedthrough mode by capturing each user's inputs and then multiplexing feedback information (replies to a single user) to the other users on the group (see Fig. 2b).

Sophisticated schemes may consider delivering less information by manipulating the granularity and timing associated with the operations executed through the groupware [22]. Interestingly, the motivation for these schemes



(a) Explicit communication (b) Feedthrough (c) Back-channel feedback

Figure 2: Groupware mediation modes. *GM* means groupware mediation and Un represents user n.

has been related to technical limitations, such as network bandwidth, but, in our point of view, the limitations of the human attentive capacity should also be accountable because the amount of information generated by the groupware may overcome us, and thus decrease group performance.

In the *back-channel feedback mode* the groupware captures unintentional information initiated by a user and directs it to another user to facilitate communication and to convey human states of attention. This may occur, for instance, when a listener says 'uh-huh' to indicate that s/he is following the speaker (see Fig. 2c). To capture this type of information the groupware interface can use attentive devices such as those described in Sect. 2, or use other sensors that take advantage of the information that is available to the groupware.

One of the concepts of the framework in Fig. 1 embraces attentive groupware devices, but first we look into some phenomena related to human attention.

3.2 Attentional phenomena

Human attention is often associated with the selection of relevant information and simultaneous attenuation or discard of non-relevant data. It is a process that optimises the use of our limited cognitive resources so that we can perceive or act accurately and quickly [23, 24, 25].

Attention and consciousness are thought to be different processes: whereas attention covers the full spectrum of data that we manipulate through our senses or memory, consciousness is confined to the information that we are currently aware of [25]. Furthermore, a recent study identifies scenarios in which attention might not give rise to consciousness and vice-versa [23]. This means, e.g., that we may be scanning a computer display without noticing important information.

Over the decades, psychologists and cognitive scientists have been identifying the goals and limitations of human attention. We argue that these findings will play an ever important role in remote collaboration, and therefore suggest that researchers should pick up this knowledge to invent and evaluate ways to support the goals and compensate the limitations.

Two of the main goals of attention are accuracy to perceive specific objects or to execute particular tasks, and speed responding to perceive objects or execute tasks after the presentation of a predictive cue [26]. Accuracy manifests itself when we successfully remove or attenuate the influence of extraneous and confusing information. The 'cocktail party' phenomenon—our ability to keep track of a conversation in a crowded room—is related to this goal [25, 24]. Speed responding occurs when we are able to respond faster to anticipated events and almost always involves a clear expectation of when to initiate the response [24].

An example of a groupware system that addresses attentional accuracy is GAZE-2 [9], which automatically regulates the sound volume of overlapping conversations according to the user's current visual focus of attention. However, the effects on group performance have not, to the best of our knowledge, been evaluated, and more contexts of group work, besides remote meetings, may benefit from explicit support of human attentional goals.

Regarding groupware support to speed responding, anticipation of upcoming events is actually quite possible in computer-mediated work because, as soon as the system detects user activity that may be pertinent to another user, it can signal that activity and also control the delivery time to create clearer expectations. For instance, popular Internet instant messengers provide anticipatory cues when users start typing a message.

Interestingly, human attention has its own limitations, in that it can take too long or even fail to select the relevant information. Moreover, some phenomena are known to occur even after tremendous training, such as the 'attentional blink,' which is a delay between paying attention to one object or task and attending to the next stimulus [27]. On the other hand, there is evidence that the response time to the second stimulus may be reduced if the time between attention switches is made longer and, in particular, constant [28]. This type of intervention should be performed by groupware systems to regulate the flows of information that each user is exposed to while collaborating.

Another attentional phenomenon is 'change blindness,' which manifests itself when we fail to notice changes, even dramatic ones, such as a swap of the person with whom we were talking to just seconds ago. As long as the change matches the context, e.g., swapping of students during a brief encounter in a university campus, we may simply miss the difference [24]. If we do want to check if anything has changed, then we may have to engage in a very slow process of scanning the full picture in front of us. This happens because, although we can attend to four or five objects simultaneously, we can only detect one change at a time [29].

The consequences of 'change blindness' in group work should be apparent. The notion of group lends itself to the creation of a social context and the existence of several people collaborating, i.e., contributing to the same shared goal, stimulates scenarios in which multiple changes may occur simultaneously. This creates the required conditions for people not noticing changes, which may reduce group performance because of the time needed to catch up. Groupware systems should, therefore, highlight changes to compensate this limitation.

3.3 Attentive devices

The last concept of our framework comprehends the input/output devices that support the groupware mediation and that provide the interface with the users. We propose a classification that comprises awareness and coupling devices, which may themselves be manipulated by specialised attentive devices.

The *awareness input/output devices* are dedicated to sensing and displaying information about collaborative activities. This information allows users to build a perceptual image of the work context and is usually designated 'group awareness' [2]. A myriad of these devices have been described in the literature [1, 2], and indeed, we argued in the motivation for the paper that groupware research has been focusing on these devices.

The *coupling input devices* are used to loose the link between the actions executed by a user and the information that is passed on to the other users [30]. Two types of coupling control may be considered: first, coupling may be exerted at the origin (on the awareness input devices) to specify what and when information about a user should become available to the rest of the group; second, coupling may be applied at the destination (on the awareness output devices) to restrict group awareness to some selected objects and actions, e.g., using filters. Note that coupling control does not apply to single-user feedback.

Coupling devices require manual discrimination and control of group awareness, thus penalising individual performance. However, this disadvantage is balanced by the capacity to limit the amount of information, which may improve attention within the group. This tradeoff sets the stage for introducing specialised *attentive devices* for groupware systems.

We propose a set of attentive devices that collect and combine information received from awareness sensors associated with each user, and that automatically manage the information that is delivered to the awareness displays, according to human attentional phenomena:

Activity anticipator (AA) Senses users' actions, or lack of activity, that may affect group performance and delivers preliminary information, to prepare users to be attentive to upcoming outcomes and to enable faster response times.

Change emphasiser (CE) Tracks awareness information available on each user's display and highlights changes caused by recent group activity, to attenuate the effects of the 'change blindness' phenomenon and to help users make faster situation assessments.

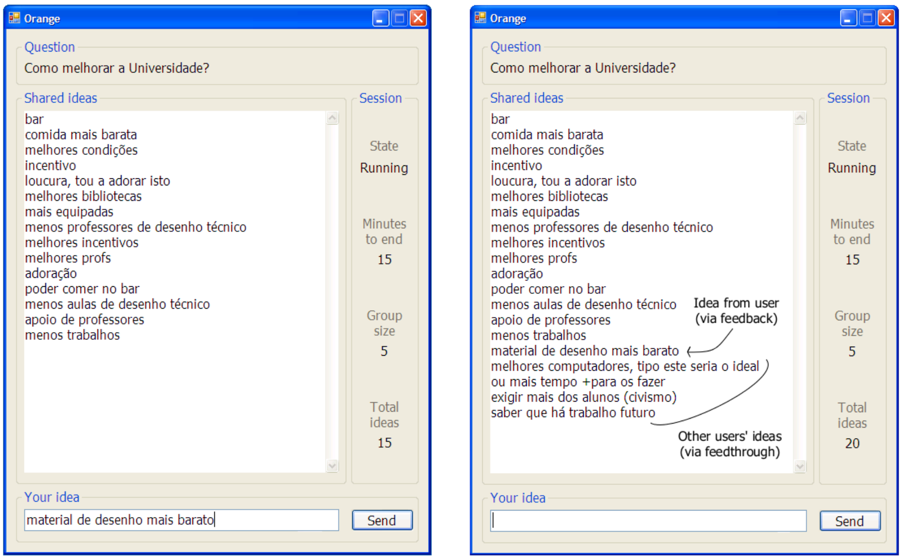
Opportunity seeker (OS) Senses activity at the user level and seeks for opportune moments to deliver group awareness, such as when s/he completes a recognisable subtask or after a brief period of inactivity, to enable faster response times to stimulus.

Time separator (TS) Intercepts the delivery of group awareness to the users, and introduces a constant delay, from the point of view of each user, after which the hand over proceeds, with the purpose of attenuating the effects of the 'attentional blink' phenomenon and to improve task switching performance.

This set of attentive devices, which may include more devices in the future, should be applicable to groupware mediation in a broad range of collaborative scenarios; for example, in asynchronous groupware, the change emphasiser may be used to highlight differences between two discrete states of group work.

4 Attentive brainstorming tool

We applied the framework for attentive groupware systems to the design of an attentive brainstorming tool, ABTool, which is currently in the second prototype.



(a) At the end of typing an idea (b) Just after sending the idea

Figure 3: Mediation exerted by the opportunity seeker in ABTool

4.1 User interface and groupware mediation

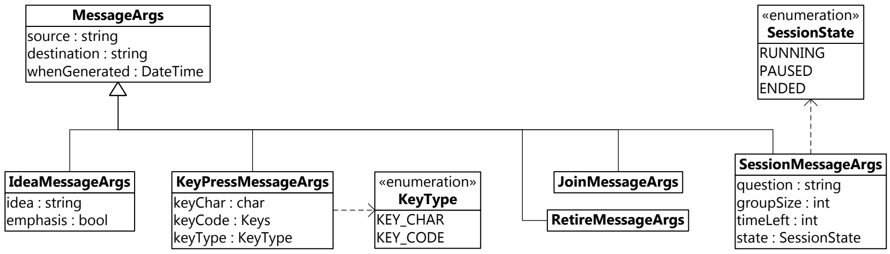
In ABTool, groupware mediation is entirely supported in the feedthrough mode, i.e., the tool automatically distributes new ideas to all users, without requiring mechanisms for explicit or back-channel communication between or among the users. All users remain anonymous during the brainstorming sessions.

In this work context, group performance depends on the number of ideas that the users produce, and this, in turn, is to some extent related to the quantity of ideas that each user is able to attend to, in particular, ideas coming from the other users. The purpose of the attentive devices is to enable users to attend to more ideas, which may have a positive net impact on group performance. ABTool currently implements three of the four proposed attentive devices:

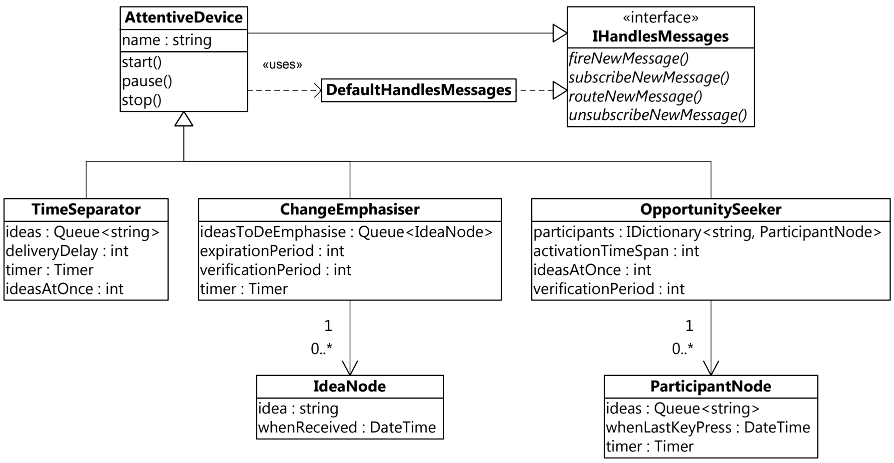
- The change emphasiser displays new ideas in bold typeface for ten seconds. When this period ends, the ideas are shown in normal typeface.
- The opportunity seeker waits for a user to finish typing an idea to then display up to ten new ideas from the other users (see Fig. 3). Alternatively, new ideas from others are also delivered if s/he stops typing for at least ten seconds.
- The time separator delivers up to ten new ideas at constant time intervals, collected over periods of ten seconds.

The two screen-shots shown in Fig. 3 were taken from the user interface available to the brainstormers, if somewhat modified to better fit side by side.

ABTool also has a user interface for the brainstorm coordinator that features three distinct areas: 1) an area showing the configuration of the current



(a) Message types



(b) Attentive devices

Figure 4: ABTool classes

session—the question, total duration, attentive device in use, session state (running, paused, or ended), and time left until it ends; 2) a similar area for configuring the next session; and 3) an area that provides a fine-grained view of what is going on in the current session, showing the group size, the total number of ideas so far, and an overview of the ideas from the users and of the feedthrough managed by the active attentive device.

4.2 Architecture and design

The attentive brainstorming tool is characterised by a client-server architecture, in which groupware mediation is performed on the server. The server also collects performance data, which is stored in an XML log. The purpose of the clients, one per user, is to receive input from the users and pass it on to the server, and to display group awareness as it becomes available from the server.

ABTool is written in C# and is built on top of the Microsoft .NET Framework 2.0. Communication between the clients and the server is done through TCP/IP sockets and all messages—ideas, key presses, users joining or retiring the group, session starting or ending (see Fig. 4a)—are automatically serialised and deserialised using BinaryFormatter objects attached to NetworkStream instances.

Within the client and server applications, messages are propagated using events, to which consumer objects can subscribe themselves. Given that almost all classes in ABTool handle message events, namely the user interfaces, the group mediator, the attentive devices, and the classes responsible for receiving and sending messages from/to the network, we defined an IHandlesMessages interface together with a default implementation for it, DefaultHandlesMessages, which relies on reflection to allow those classes to delegate the determination of the method to run as a function of the type of message associated with the event.

Figure 4b shows that the attentive devices in ABTool derive from the AttentiveDevice generalisation, which actually implements instant feedthrough, i.e., with this device all ideas from each user are immediately forwarded to the group. The TimeSeparator device introduces a delivery delay, supported by a Timer object, during which ideas are stored in a queue. At the end of the delay, up to ideasAtOnce ideas (default is ten) are delivered to the group.

The ChangeEmphasiser has a queue of currently emphasised ideas, each one with an attached time-stamp, and every verificationPeriod milliseconds (default is one second), removes the emphasis of those ideas that have been put forward more than expirationPeriod seconds ago.

Finally, the OpportunitySeeker device maintains separate queues, one for each user, containing ideas that have been put forward by the other users on the group. The queue is stored in the ParticipantNode, which also keeps a Timer object that every verificationPeriod milliseconds verifies the time of the most recent key press by the user, and if it was more than activationTimeSpan milliseconds ago (default is ten seconds), then it delivers up to ideasAtOnce ideas to the user.

The attentive devices implement three methods: start() is run when a session starts or resumes; pause() is executed when, for some reason, the session needs to be paused; and stop() is run at the end of a session. Other methods handle the reception and forwarding of messages, but we omitted those for brevity.

5 Evaluation

We now describe a laboratory experiment that we set up to evaluate the impact on group performance caused by the use of the attentive devices in the context of electronic brainstorming, using ABTool, and present preliminary results.

5.1 Participants

A total of 25 volunteers (17 men and 8 women), organised in groups of 5, successfully participated in the experiment. The median age of the participants was 23 (M=23.8, minimum 20, maximum 29). A convenience sampling was used to select participants, who were recruited from social contacts, from posters on corridors, and from adverts shown in classes at the Faculty of Sciences of the University of Lisbon. No monetary reward was offered and the only information available was that the experiment would concern brainstorming. 21 participants were students (15 undergraduate, 5 MSc, 1 PhD), and the remaining 4 comprised researchers, a software developer, and a translator. This method for population sampling may introduce bias due to the specialisation of the participants, but it is a compromise between available resources and generalisation of results, which is also evident in about 90% of the research done with groupware [31].

(a) Group of participants (b) Detail of work place

Figure 5: In the laboratory room

5.2 Apparatus

The experiment was conducted in a laboratory room (illustrated in Fig. 5) having five Sony Vaio TX laptops with identical hardware (Intel Pentium M at 1.2 GHz, 1 GByte of RAM) and software (Microsoft Windows XP SP2), interconnected by a local 100 Mbit/s ethernet network. Keyboard sensitivity, desktop contents, screen resolution, and brightness were controlled. Each computer had screen-recording software (ZD Soft Screen Recorder 1.4.3), and a web-camera (Creative WebCam Live!) affixed to the top of the display. The client application of ABTool was installed on the five laptops, while the server was running on an IBM ThinkPad T40 (Intel Pentium M at 1.3 GHz, 1.25 GBytes of RAM).

5.3 Task

Participants completed practice and test tasks, both related to brainstorming. The practice task enabled participants to get acquainted with ABTool. In the test task participants were asked to generate as many ideas as possible by typing on the keyboard and by looking at the computer screen; speech or other forms of communication were disallowed.

5.4 Design

A single-factor, repeated measures design was chosen for the experiment. Four treatments—one control and three experimental—were applied, where each experimental treatment corresponds to an attentive device (the independent variable). The dependent variable, group performance, was operationalised from the sum of the number of ideas produced by each user on the group on each brainstorming session. All groups received the four treatments, and the order in which each group of participants was exposed to the treatments was balanced using a latin square, which ensures that every treatment is first, second, and so forth, only once per sequence of four trials. We also used a latin square to determine the order of the brainstorming questions in each trial.

5.5 Procedure

The experiment was held during a full week in March 2007 and took a total of approximately twelve hours to run, including about fifty minutes for maintenance tasks each day. Each experimental trial took about one hour and thirty minutes: the practice task lasted for five minutes, and each test task took fifteen minutes, followed by a five minute rest period. A trial started when a group of participants arrived at the laboratory room. An introduction to this research was given, and participants were informed on their privacy rights and asked to sign a consent form. Next, participants filled in an entrance questionnaire about gender, age, and type (e.g., student). Written instructions on the brainstorming rules and on the ABTool application was then handed in to each participant.

Participants were asked to carry out the practice task, after which questions about ABTool were answered. The group then performed the first test task. Next, during the rest period, participants filled in a self-subjective questionnaire, which is out of scope for this paper. This arrangement of test task and rest period was done for all four treatment conditions as defined in the latin square.

At the end of the trial, answers were given to the questions participants had about the research, and the experimenter gave thanks in acknowledgement of their participation in the experiment.

5.6 Analysis

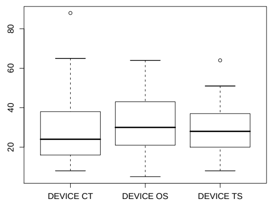
Results encourage further evaluation with the opportunity seeker attentive device, which showed a median of 25 more ideas out of scores ranging from 83 to 264 ideas per group per session. Considering the whole experiment, the difference was 790 ideas with the opportunity seeker against 723 with the control device. On the other hand, the non-parametric Wilcoxon matched-pairs signed-ranks test [32] gave a 15.6% probability of chance explaining the difference in scores between the opportunity seeker and the control treatment, which is not statistically significant at the usually accepted 5% α-level.

Regarding the contrast between group performance with the time separator and the control treatment, the p-value for the same Wilcoxon test was 59.4%, which reflects the non-difference of only two ideas in the total number of ideas for the whole experiment.

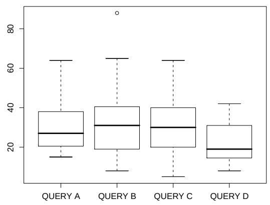
We did not include the change emphasiser treatment in the analysis because we detected a problem with this device: it did not deemphasise ideas due to an unintended dependency between the client and server clocks. This problem was fixed afterwards.

We consider these results as preliminary and interpret them with caution due to the low number of available scores and also because the non-parametric Friedman rank test for k correlated samples [32] suggests that there is no significant change in group performance caused by the mediation exerted by the attentive devices ($\chi^2_F = 1.2$ on 2 *df*, $p = 0.55$).

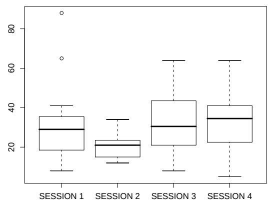
We also intended to analyse the hypothetical influence of session order and brainstorming question in group performance but because of the removal of the change emphasiser treatment, the experimental design became unbalanced, i.e., there are different number of scores in terms of the session number or the brainstorming question, and so the Friedman test cannot be applied. We did not anticipate this type of situation when we chose the repeated measures design.



(a) by treatment



(b) by question



(c) by session number

Figure 6: Number of ideas per user per session

At this point we decided to also analyse performance at the user level since we have about five times the number of scores and also because it seems plausible that group performance in brainstorming sessions is linked to individual performance, at least to some extent.

Figure 6 shows the scores in terms of treatment, brainstorming question, and session number. We note that in Fig. 6b, questions A and D have only 15 scores (out of 25 for query C), and that in Fig. 6c there were only 15 data points for session 2 (out of 20 for the other session numbers).

The median number of ideas per user per session was 24 for the control treatment (CT), 28 for the time separator (TS), and 30 for the opportunity seeker (OS). The Wilcoxon matched-pairs signed-ranks test [32] on CT and TS detected no difference in scores, with $p = 34\%$, but the contrast between the control and opportunity seeker treatments was more relevant, almost achieving statistical significance at the 5% α-level, with a p-value of 5.4%. These results reinforce our interest in further evaluating the opportunity seeker attentive device.

6 Conclusions and future work

The study of the mediating role of computers on human attention in remote collaboration settings is largely an unexplored research area. While current trends keep aiming at conveying ever greater awareness information about the group, we suggest a route that explicitly recognises the limitations of the human, and therefore the group, attentive capacity. This route is consistent with AUI research and we argue that the existing body of knowledge should extended into the groupware field. To this end we introduce a framework for attentive groupware systems and hypothesise that group performance improves with the use of specialised attentive devices.

Many questions remain unanswered: does group performance significantly improve? What attentive devices are best suited for different collaborative scenarios? Can groups be made larger while remaining attentive and productive? We are addressing some of these questions with an attentive brainstorming tool, ABTool, and setting up laboratory experiments where we measure the number of ideas produced by the group in terms of each attentive device, using multiple comparisons with repeated measures and non-parametric statistical analysis.

The obtained results so far indicate that the opportunity seeker—which waits until the user finishes typing an idea to then deliver the ideas from the other users on the group—is more effective, which is in line with some of the conclusions by

Bailey and Konstan [17] with singleware systems. The other attentive device that we successfully experimented with, the time separator, had no significant impact on group performance. The experiments will carry on with the inclusion of a third device, the change emphasiser, to further explore the use of attentive devices for groupware systems.

Acknowledgments

We thank Pedro Custódio and Bruno Coelho for their assistance during the preparation and execution of the experimental trials.

References

[1] Gitesh K. Raikundalia and Hao Lan Zhang. Newly-discovered group awareness mechanisms for supporting real-time collaborative authoring. In *AUIC'05: Proceedings of the sixth Australasian conference on User interface*, pages 127–136, Newcastle, Australia, January 2005.

[2] Carl Gutwin and Saul Greenberg. A descriptive framework of workspace awareness for real-time groupware. *Computer Supported Cooperative Work*, 11(3):411–446, 2002.

[3] Roel Vertegaal. Attentive user interfaces. *Communications of the ACM*, 46(3):30–33, 2003.

[4] Claudia Roda and Julie Thomas. Attention aware systems: Introduction to special issue. *Computers in Human Behavior*, 22(4):555–556, 2006.

[5] Roel Vertegaal, Jeffrey S. Shell, Daniel Chen, and Aadil Mamuji. Designing for augmented attention: Towards a framework for attentive user interfaces. *Computers in Human Behavior*, 22(4):771–789, 2006.

[6] Eric Horvitz, Carl Kadie, Tim Paek, and David Hovel. Models of attention in computing and communication: From principle to applications. *Communications of the ACM*, 46(3):52–59, 2003.

[7] James Fogarty, Andrew J. Ko, Htet Htet Aung, Elspeth Golden, Karen P. Tang, and Scott E. Hudson. Examining task engagement in sensor-based statistical models of human interruptibility. In *CHI'05: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 331–340, Portland, Oregon, USA, April 2005.

[8] Patrick Baudisch, Doug DeCarlo, Andrew T. Duchowski, and Wilson S. Geisler. Focusing on the essential: Considering attention in display design. *Communications of the ACM*, 46(3):60–66, 2003.

[9] Roel Vertegaal, Ivo Weevers, Changuk Sohn, and Chris Cheung. GAZE-2: Conveying eye contact in group video conferencing using eye-controlled camera direction. In *CHI'03: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 521–528, Ft. Lauderdale, Florida, USA, April 2003.

[10] Tracy Jenkin, Jesse McGeachie, David Fono, and Roel Vertegaal. eyeView: Focus+context views for large group video conferences. In *CHI'05: Extended abstracts on Human factors in computing systems*, pages 1497–1500, Portland, Oregon, USA, April 2005.

[11] Roel Vertegaal. The GAZE groupware system: Mediating joint attention in multiparty communication and collaboration. In *CHI'99: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 294–301, Pittsburgh, Pennsylvania, USA, May 1999.

[12] Shumin Zhai. What's in the eyes for attentive input. *Communications of the ACM*, 46(3):34–39, 2003.

[13] Aulikki Hyrskykari, Päivi Majaranta, Antti Aaltonen, and Kari-Jouko Räihä. Design issues of iDICT: A gaze-assisted translation aid. In *ETRA'00: Proceedings of the 2000 symposium on Eye tracking research and applications*, pages 9–14, Palm Beach Gardens, Florida, USA, November 2000.

[14] Shamsi T. Iqbal, Piotr D. Adamczyk, Xianjun Sam Zheng, and Brian P. Bailey. Towards an index of opportunity: Understanding changes in mental workload during task execution. In *CHI'05: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 311–320, Portland, Oregon, USA, April 2005.

[15] Daniel Chen and Roel Vertegaal. Using mental load for managing interruptions in physiologically attentive user interfaces. In *CHI'04: Extended abstracts on Human factors in computing systems*, pages 1513–1516, Vienna, Austria, April 2004.

[16] Dennis W. Rowe, John Sibert, and Don Irwin. Heart rate variability: Indicator of user state as an aid to human-computer interaction. In *CHI'98: Proceedings of the SIGCHI conference on Human factors icomputing systems*, pages 480–487, Los Angeles, California, USA, April 1998.

[17] Brian P. Bailey and Joseph A. Konstan. On the need for attention-aware systems: Measuring effects of interruption on task performance, error rate, and affective state. *Computers in Human Behavior*, 22(4):685–708, 2006.

[18] Daniel C. McFarlane. Comparison of four primary methods for coordinating the interruption of people in human-computer interaction. *Human-Computer Interaction*, 17(1):63–139, 2002.

[19] António Ferreira and Pedro Antunes. Dispositivos de gestão da atenção em sistemas colaborativos. In *Interacção'06: Actas da segunda conferência nacional em Interacção pessoa-máquina*, pages 57–60, Minho, Portugal, October 2006.

[20] Pedro Antunes, António Ferreira, and José A. Pino. Analyzing shared workspaces design with human-performance models. In *CRIWG'06: Proceedings of the twelfth international workshop on Groupware*, pages 62–77, Medina del Campo, Spain, September 2006.

[21] Jason Hill and Carl Gutwin. Awareness support in a groupware widget toolkit. In *GROUP'03: Proceedings of the 2003 international ACM SIGGROUP conference on Supporting group work*, pages 258–267, Sanibel Island, Florida, USA, November 2003.

[22] Carl Gutwin and Saul Greenberg. The effects of workspace awareness support on the usability of real-time distributed groupware. *ACM Transactions on Computer-Human Interaction*, 6(3):243–281, 1999.

[23] Christof Koch and Naotsugu Tsuchiya. Attention and consciousness: Two distinct brain processes. *Trends in Cognitive Sciences*, 11(1):16–22, 2007.

[24] John Anderson. *Cognitive Psychology and its Implications*, pages 63–105. Worth, sixth edition, 2005.

[25] Robert J. Sternberg. *Cognitive Psychology*, pages 65–107. Wadsworth, third edition, 2003.

[26] David LaBerge. Attention. In Benjamin Martin Bly and David E. Rumelhart, editors, *Cognitive Science*, pages 44–98. Academic Press, San Diego, CA, USA, 1999.

[27] Michael W. Eysenck and Mark T. Keane. *Cognitive psychology: A student's handbook*, pages 119–150. Psychology Press, fourth edition, 2000.

[28] Claudia Roda and Julie Thomas. Attention aware systems: Theories, applications, and research agenda. *Computers in Human Behavior*, 22(4):557–587, 2006.

[29] Daniel J. Simons and Ronald A. Rensink. Change blindness: Past, present, and future. *Trends in Cognitive Sciences*, 9(1):16–20, 2005.

[30] Prasun Dewan and Rajiv Choudhary. Coupling the user interfaces of a multiuser program. *ACM Transactions on Computer-Human Interaction*, 2(1):1–39, 1995.

[31] Jerry Fjermestad and Starr Hiltz. An assessment of group support systems experimental research: Methodology and results. *Journal of Management Information Systems*, 15(3):7–149, 1999.

[32] David C. Howell. *Statistical methods for psychology*, pages 635–664. Thomson, sixth edition, 2007.