ATGENTIVE

IST-4-027529-STP

DELIVERABLE D1.3

Status: Public
VERSION: V 1

Atgentive Conceptual Framework and Application Scenarios

THE AMERICAN UNIVERSITY OF PARIS
knowledge, perspective, understanding

AUP, France, September 12, 2006
Claudia Roda (Editor)

The following people have contributed to this deliverable:

AUP Damien Clauzel Claudia Roda Marco Raglianti Georgi Stojanov	**Cantoche** Laurent Ach Benoit Morel	**CELN** Jaroslav Cech	**INSEAD** Thierry Nabeth *(technical coordinator)* Pradeep Kumar Mittal Ye Deng
OBU Paul Rudman Mary Zajicek	**Ontdeknet** David Kingma Inge Molenaar Koen Molenaar Maurice Vereecken	**UTA** Kari-Jouko Raiha Harri Siirtola Toni Vanhala Kimmo Vuorinen Veikko Surakka	**STC**

Atgentive conceptual framework and application scenarios

Abstract

This deliverable analyses attention in learning and collaboration with the aim of defining a theoretical framework capable of representing the wide range of attentional processes that one may want to support within learning and collaboration environment. The analysis aims at being exhaustive rather than selective: in future deliverables a small subset of this framework will be selected for possible implementation.

We start by proposing that the learning processes may be supported at several different levels (regulative, cognitive, and meta-cognitive) and that such support may be translated in a set of corresponding interventions directing the learner attention to the appropriate foci. We further propose that the process of *directing attention* is in itself decomposable in a set of levels (perception, deliberation, operation, meta-cognition) allowing one to support the corresponding attentional processes. We substantiate this proposal with a set of realistic scenarios exemplifying the theoretical framework. We then discuss how the theoretical framework may be modelled in an event based conceptual framework for the support of attentional processes in learning. This conceptual framework is demonstrated by discussing how the realistic scenarios previously introduced would fit the event-based model proposed.

Table of Contents

1 Introduction

This deliverable proposes the conceptual framework for the Atgentive project and it draws on several technical notes produced by project partners during the first six month of our work together.
Although we have aimed at making this document self-contained, some knowledge of the project's Description of Work, and previous deliverables (especially (Roda, 2006; Rudman & Zajicek, 2006; Stojanov & Roda, 2006)) will help in the reading of this deliverable. For the reader with no previous knowledge of the project we include in appendix 1 a recent publication that gives an overview of the project (Roda & Nabeth, 2006).

The key hypothesis behind Atgentive is that intelligent agents - whether accompanied by an embodied, language, or cue based representation - may support learners and knowledge workers in two key manners:

- First, agents may support users in their attentional choices, by for example, helping them recollecting essential information when resuming an interrupted task, or providing help in situations of impasse.
- Second, agents may act as guides steering the user's focus towards foci more relevant to their long and short term goals. For example, agents may notify users of important events, remind them of deadlines, or suggest alternative activities for goal achievement.

The conceptual framework proposed in this deliverable aims at providing a theory capable of describing the many different aspects of attention support in learning and collaboration. For this reason, the analysis is exhaustive rather than selective. Obviously, not all the scenarios, and intervention strategies discussed here will be included in the overall system design that the consortium will propose. In future deliverables a small subset of this framework will be selected for possible implementation on the basis of two main selection criteria: feasibility and desirability. Feasibility will be evaluated on the basis of the consortium's technical and human resources; desirability is being and will be evaluated through the formative evaluation, see (Rudman & Zajicek, 2006).

This chapter gives a general introduction to the support of attention in learning and collaboration as they may be implemented in systems capable of supporting human attentional processes. These systems are referred to as *attention aware systems* (Roda & Thomas, 2006). We propose that the learning processes may be supported at several different levels (regulative, cognitive, and meta-cognitive) and that such support may be translated in a set of corresponding interventions directing the learner attention to the appropriate foci. In chapter 2 "A classification of possible levels of attention support" we analyse how attention may be supported at four different levels: perceptual, deliberative, operational, and meta-cognitive; we substantiate this "top-down" analysis of attention-aware-systems with a set of realistic scenarios exemplifying the theoretical model. Chapters 3 and 4 analyse how support at the levels identified in the theoretical model of chapter 2 may be realised in an event based conceptual framework for the support of attentional processes. This conceptual framework is demonstrated through the instantiation of the same scenarios presented in chapter 2 as applied to the two Atgentive demonstrators systems: AtgentNet and AtgentSchool.

The proposed conceptual framework is meant to be relevant for a wide range of systems. We believe that such widespread relevance is demonstrated by the applicability of our theoretical models, conceptual framework, and scenarios, to the AtgentNet and AtgentSchool pilots. The two pilots, in fact, differ in several significant manners:

- Their main objectives and user groups: AtgentNet aims at supporting collaboration amongst professional adults, AtgentSchool aims at fostering learning amongst young children.
- The assumptions they make in supporting attention: AtgentNet is characterised by the fact that in general the system will not be able to detect the user's goal, as is instead the case in the AtgentSchool environment.
- Their attention allocation strategies: Attention allocation with respect to collaboration is the main focus in AtgentNet. Attention allocation with respect to learning is the main focus in AtgentSchool.
- Their attention support modalities: In the AtgentNet application the prevalent modality is "attention support at the interface"; in the AtgentSchool application the prevalent modality is "attention support at the agent level". This difference is mainly due to the different

environments and user groups. In AtgentNet the system mainly aims at presenting information in a manner that facilitates the user's decisions on how to allocate his/her attention. In AtgentSchool the system mainly acts as a guide for attention allocation.

- Attention related information available: AtgentNet must base its reasoning on largely de-contextualised input sequences of the user (users often use the system irregularly, and with varying objectives). AtgentSchool is often capable to relate user's input sequences to specific learning activities and reason about the user's learning behaviour.

1.1 Learning and attention

This section aims at connecting learning theory and attention theory by considering how supporting certain attentional foci, problems that occur in the learning process may be solved.
Previous research (the details of which are reported in appendix 2) has allowed us to classify learning problem as occurring at three possible levels: regulative, cognitive and/or meta-cognitive. The three problems can be connected to three intervention levels in order to redirect attention to the right stimuli to continue or improve the learning process.

Various authors have already suggested that the learning process consists of different layers that are intertwined with each other. The distinction among the layers has been described in many ways. Nelson (Nelson, 1996) has divided learning in two levels: the object-level and the meta-level. The object level entails the actions and behaviours of he learner and all information that is relevant for the context at hand. The meta-level entails dynamically assessing the present situation. Control and monitoring information flow from the meta- to the object-level. This model is very useful to differentiate between cognition and meta-cognition.

Following the study reported in appendix 2 we have added one level by making a division between the actions and behaviours of the learner and the information that is relevant for the task. This leads to the division into three levels: regulation, cognition and meta-cognition. Below these levels will be explained in more detail. The arrows in figure 1 show the heavy intertwinement of these levels.

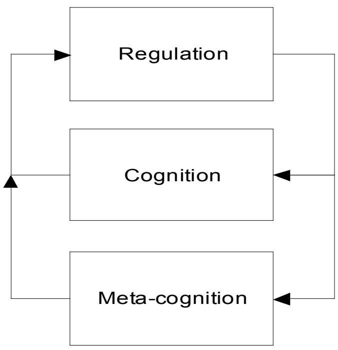


Figure 1. Classification of learning problems

1.1.1 Regulation

The regulation of the learning process deals with the actions and behaviours of the learner at the lowest level. It supports the physical actions a learner has to perform to support the learning. A simple example of a regulation activity is moving closer to the screen to be able to better read the text.
The student will typically execute a regulative activity automatically. When the activity can not be solved automatically the learner will become consciously aware of it.

1.1.2 Cognition

Cognition deals with the content and context of the learning task. Cognition is situated between regulation and meta-cognition. It is defined as the knowledge and skills that are necessary to perform the task (Garner, 1987). A simple example of cognition is the knowledge a learner has about how one may introduce himself/herself to perform the task, required by the e-learning application, to introduce himself/herself.

The attention of the students at occurrence of a cognitive activity will be consciously directed at the task at hand. The first focus at a cognitive level is to asses the needed information to perform the task.
When a student is unable to asses the needed information a strategy is needed to acquire the needed information. When the student has enough self-regulations skills they might be able to assess the right strategy / process to perform this cognitive activity themselves. This is where meta cognition comes into the picture.

1.1.3 Meta cognition

Meta cognition is necessary in order to understand how a task can be performed; it is situated at the high end of the learning process and has a very strong relation with cognition as described above. The meta cognitive knowledge and skills are the ability to understand where you are in your learning process and how you should continue. It entails the episodes shown in figure 2:

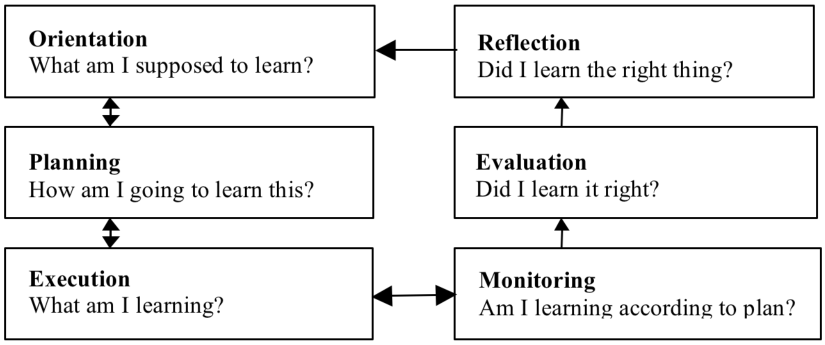


Figure 2. meta-cognition

Attention of student dealing with a meta-cognitive activity is consciously directed at the current process they need to reflect on.

1.1.4 Interventions supporting attention in learning

In essence we can distinguish between students that are actively directing their conscious attention to the learning task and the students that are not paying attention to the learning task. Teachers continually assess the student attention and come to the conclusion whether a student is directing his attention in the right way.
When a student is not paying attention, the attention of the student is directed at the wrong stimuli. This typical relates to a regulative problem. This distraction is often caused by a stimulus attracting the attention of the student to something different than the task and typically triggers off task student behaviours.
A student is on task when the attention is direct at the stimuli supporting the learning process. The student is using a conscious or top-down, goal-driven attention processes in order to learn. Cognitive and meta cognitive problems can be experienced to continue their learning. An interference in these situations situation will be directed at the conscious or top-down, goal-driven attention of the users.

The following three intervention rules can be deduced from this framework:

- When the student is facing a regulative problem, a simple procedural interference will be necessary to redirect the learner's attention.
- When a student is facing a cognitive problem, an intervention directing attention to the information necessary to perform the task will be needed.
- When a student is facing a meta-cognitive problem, an intervention directing attention at the learning process is needed.

The distinction between the different layers in the learning process can support the analysis of the situation at hand and the useful intervention. Regulative problems are most likely caused by a pre-conscious or bottom up attention switch with students distracted by stimuli in their surroundings. Conscious or top-down attention shifts are more likely to occur while students are confronted with cognitive of meta-cognitive problems.
This defines three intervention possibilities, namely a procedural intervention following a regulative problem, a content intervention following a cognitive problem and a process intervention following a meta-cognitive problem. Based on the assessment of the situation at hand the teacher or Atgentive can select the right intervention modus.

1.2 Collaboration and attention

In the context of virtual communities, attention aware systems may reduce information overload and help learners in better managing interruptions by providing them with a set of mechanisms helping:

- to better perceive the environment (filtering irrelevant information, enhancing important one);
- to improve their learning and working practice in a way that is more attention effective (by helping them to better assess and manage the way they are allocating their attention);
- to reduce the cognitive effort required to accomplish their tasks (by simplifying or automating their processes).

As it will result obvious from the discussion below, one important aspect of attention support is personalisation. Research in cognitive psychology has demonstrated that most human beings have some basic reactions that make them sensitive to certain stimuli (e.g. luminance change). In the general case however, each learner will perceive different stimuli in a very personal way, that will depend on personal characteristics as well as on the specific situation, learner's goal, and environmental conditions. Whilst it is possible for the system to completely adapt its interface and behaviour to each user, in the case of multi-user interaction, this adaptation can reduce the shared perception of the group. For instance, the finding of a common understanding would appear difficult to achieve between two or more individuals that would have a different interface to the system and therefore would perceive the system in a totally different way. The concept of the magic lens (Bier, Stone, Pier, Buxton, & DeRose, 1993) can be used to address this issue by enhancing the perception of the individual without doing it at the detriment of the perception of the group. Practically, a magic lens can provide to each user a slightly customized perspective of the same, shared representation of the information. Hence, with a magic lens, two different users will be able to observe the same spaces and the same items. However, the Magic lens will provide to each user “wearing this lens” a magnified (or on the contrary reduced) perception of the items that are the more relevant for him/her, and therefore a slightly different view of the environment. See also Deliverable 2.1 "State of the Art” (Roda, 2006) for a more in-dept presentation of the support of attention in a working context.

1.3 From the support of learning and collaboration to the support of attention

In section 1.1 we have proposed a theoretical model describing the various levels at which it is possible to support "learning" processes. That model addresses the following two questions:

- What type of learning problems may the student encounter? (at regulation, cognition, meta-cognition level)
- "To what" should attention be redirected in order to address the problems? (procedural intervention, intervention directing attention to the information necessary to perform a task, intervention directing attention to the learning process)

In the rest of this document we consider the following question:

- How can attention be redirected to the desired focus?

We propose a theoretical model for the support of attention processes acting at four levels (perceptual, deliberative, operational, meta-cognitive level) and we analyse several realistic scenarios to exemplify our theory.

Figure 3 below, describes the bridge between the two models (learning support, and attention support) illustrating how one may reason in order to guide the learner attention by first identifying the learning problem and the desired focus of attention, and then intervening at the appropriate level.

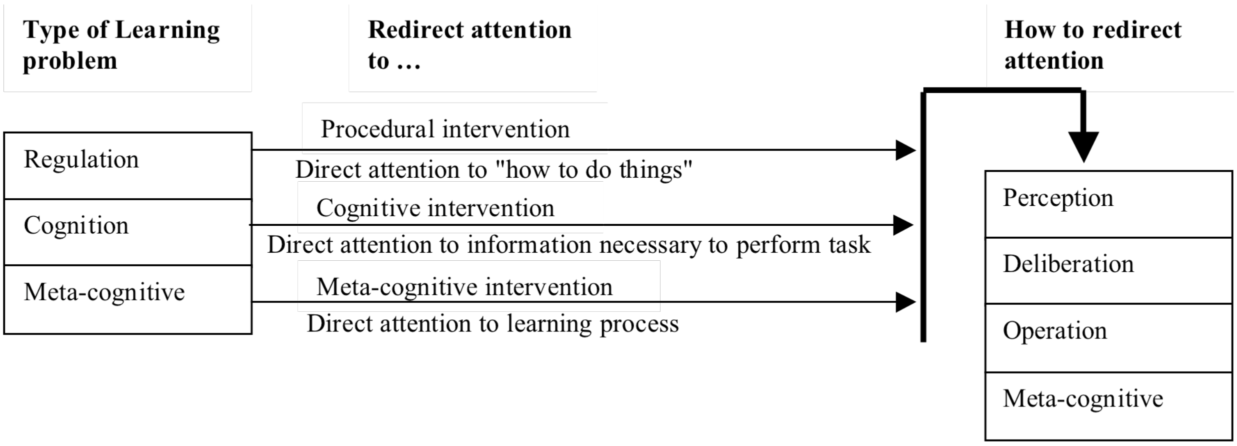


Figure 3 – Guiding learners' attention.
Learning support model (on the left) and attention support model (on the right)

2 A classification of possible levels of attention support

Studies in cognitive psychology have clearly established (see for example review in (Roda & Thomas, 2006)) that attention allocation depends both on perceptual and deliberative processes. At the **perceptual level**, what we perceive (e.g. see, hear, feel) impacts on what we pay attention to. At the **deliberative level** our goals, motivations, and intentions, also play a role in the determination of our attention focus. Furthermore, some of our actions may require a varying degree of cognitive effort depending on how easily we can focus on the activity. This implies that at **operational level** the environment in which action takes place determines how easily we can attend to a given focus. Finally, we are able, as we are doing here, to reason about our own strategies for allocating attention. At the **meta-cognitive level** one may elaborate and evaluate strategies for attention allocation that are, or will be, implemented at the cognitive level. Following these principles, this deliverable analyses how to support attention:

- At the **perceptual level** by, for example, facilitating access to relevant information or presenting interruptions at the appropriate level of conspicuity. See section 2.1
- At the **deliberative level** by, for example, supplying tools for the control of task priorities, or by motivating users who are loosing focus. See section 2.2
- At **operational level** users may be supported by simplifying some attention related operations such as restoring the context of interrupted tasks, or by filtering incoming messages. See section 2.3
- Finally, at the **meta-cognitive level** the users can be supported in their reflection about how they allocate attention by, for example, providing self-diagnostic tools. See section 2.4

This division is obviously artificial and only finalised to guide our analysis since it is always a combination of the above factors that intervene in attention allocation. A good example of how factors at different levels are interrelated is supplied by one of the most researched aspects of attention support: interruption and notification management. In the detailed description of attention support at the four levels, provided below, scenarios involving interruption management appear at each one of these levels. Note that, along the lines already proposed in (Roda & Thomas, 2006) we see our framework as an extension of *attentive dispatching* (notification) systems and in particular, notification management is only one aspect of the support provided in fact we assume that: "(1) support may be given to users in orienting their attention as well as in maintaining it on a current task. (2) Users may overlook important information that is already available and they may want to be helped in the selection of this information. (3) Attention switches initiated by the user or by events in the environment are just as relevant as the attention switches provoked by the system, and should be evaluated with respect to the user goals." (Roda & Thomas, 2006, p. 578).

In this chapter support at each level introduced above (perceptual, deliberative, operational, and meta-cognitive) is analysed and exemplified by a set of scenarios[1]. These scenarios will be further developed in chapter 4 and they will be applied to the two Atgentive's pilots: AtgentNet and AtgentSchool. A summary of our classification, along with the scenarios is shown in section 2.5, Table 1.

2.1 Supporting attention at the perceptual level

Supporting perception means increasing both the ability to notice relevant information and to discard irrelevant one. We recognise at least four different manners in which perception may be enhanced: (1) facilitating the selection of relevant information (2) facilitating information comprehension, (3) supporting group perception, and (4) presenting interruptions at the correct level of conspicuity[2].

2.1.1 Facilitating the selection of relevant information

As suggested by many authors, information selection may be facilitated by the use of **information filtering mechanisms**. Such mechanisms allow the user to select, statically (i.e. once for all) or dynamically, relevant types of information. Filtering mechanisms may be based on more or less sophisticated selection algorithms or, in the case of social filtering, may use human evaluators who classify or assign ratings to various items. Information filtering may result in avoiding presentation to the user of certain items (as in the case of spam filters filing messages classified as spam in a "spam folder" rather than in the "inbox") or in making more noticeable the information deemed to be more relevant. Although any of the human communication channels can be used with the aim of providing information at various levels of prominence, the most studied channel (and therefore the one we know most about) is the visual one.

Appropriate **visualisation** of information items however, is not as straightforward as it may seem (Healey, 2005). Many cognitive factors play a role in what will attract our attention so that, in order to "make something more visible" it is necessary to consider how we, human beings, select visual information. On the one hand research on preattentive processing such as Treisman's *feature integration theory* (Treisman, 1998; Treisman & Gelade, 1980) demonstrate that certain searches on the visual field may be made very efficient if they only involve *basic features*, i.e. features that can be extracted preattentively (for instance finding the only red object in a very complex scene; see also (J. M. Wolfe, 2001)). On the other hand, it is clear that our ability to "see" something does not depend solely on what we are presented with, but also on what we are looking for, or we expect (Egeth & Yantis, 1997; Rensink, 2000; J. Wolfe, Klempen, & Dahlen, 2000).

Many visualisation systems and techniques address, more or less explicitly, attentional processes in the visual modality. Toet (2006) reviews attention related adaptive techniques for visual information presentation. Abowd, Mynatt, and Rodden (2002) review also non-visual modalities that have been studied in interfaces for intelligent environments and ubiquitous computing where several channels are often employed in order to communicate with the user. Bearne, Jones, and Sapsford-Francis (1994, p. 105) proposing guidelines for the design multimedia systems, discuss how different attention modalities (e.g. auditory and visual) may be integrated in a system whilst avoiding interference between the different types of interaction. More generally, good design principles for creating systems facilitating the identification of relevant information by the user (e.g., amount of text on pages, numbers and types of links, consistency, accessibility, etc.) have been extensively studied under the domain referred as Information Design (Beier & Vaughan, 2003; Ivory & Megraw, 2005).

In certain situations information may be made more visible, credible, and comprehensible by being communicated by an **embodied agent** (Lester, Converse, Kahler, Barlow, Stone, & Bhogal, 1997; Picard, 1997, 2004). Many factors may influence the user's reaction to information presented by such embodied agents, these include the agent's appearance, voice, size, etc. We have already presented a short analysis of these factors in section 7 of deliverable D1.1 (Roda, 2006). In our model we assume an "Atgentive embodied agent language" that allows embodied agents to react to system requests such as "Give positive feedback", "Give negative feedback", "Ask a question", "Suggest", "Agree", "Disagree", "Reply". These requests will possibly be accompanied by modifiers such as: "Emphatic", "Normal", "Subdue". Requests and modifier will help the embodied agent decide how to speak, present some text, or just use body language.

[1] The scenarios presented in this deliverable have been produced and elaborated on during the course of the project and then organised following the theoretical and conceptual framework presented here. Several Atgentive's internal documents refer to these scenarios by chronological (by creation) number. To maintain consistence with those documents the numbers of the scenarios have not been changed here; for this reason the scenarios numbering is not sequential in this document.

[2] Conspicuious: obvious to the eye or mind. The Merriam-Webster Online Dictionary

2.1.2 Facilitating information comprehension

Several techniques have been, or may be, used to facilitate information comprehension and therefore reducing the cognitive load associated to selecting the most relevant pieces of information, or the most appropriate action to perform. They involve providing **meta-level** / abstract information about documents or tasks. For instance, indicators that abstract the information related to a document may include the category of the document, the size of the document, the date of creation of the document, the author, but also the popularity of the document (how many people have accessed the document). The indicators that abstract a discussion space may include the number of threads, the number of messages that have been posted in the last day, or the topics generating most debate. A recent project addressing the collection of attention related meta-data is attention. XML (Sifry, Marks, Çelik, & Hayes, 2006) which targets blogs and feeds and tracks data such as: what has been read, what the user has spent time on, recommendations, etc. Abstract information may also display relationships amongst documents or tasks in the form of graphs allowing users to quickly understand and navigate a web of interrelated items. Finally, the use of metaphoric or iconic tags allows, in certain cases, to abstract essential properties of documents facilitating the immediate grasps of the document type or its content (an email, an email address, a document about a certain project, etc.)

Supplying meta-level information contributes to complexity reduction (by displaying selected aggregated data) and diminishes the effort required by the user to collect the information. This concept of supplying meta-data information is similar to the concept of magic lens that has been introduced by Bier and al. (Bier, Stone, Pier, Buxton, & DeRose, 1993, p.73) as “filters that modify the presentation of application objects to reveal hidden information, to enhance data of interest, or to suppress distracting information”.

Finally, abstracting information can also help to reduce the number of chunks of information that is manipulated by the user and in particular keep this number under the 7 limit imposed by the limitation of the human short term memory (Miller, 1956).

2.1.3 Supporting group perception

When working or learning in a group, the group's activity influences the focus of attention of each individual in several manners. The following are three examples of such influence. First, devoting one's attention to a given activity may be necessary in order to ensure the good functioning of the whole group (as in the case when one allocates attention to a task because the completion of the task is a prerequisite for the activity of other members of the group). Second, mechanisms of peer pressure may result in diverting one's attention from its natural course (as in the case when one looks in a certain direction because he notices that several members of the group are looking in that direction). Third, one's decision on whether to interrupt the activity of others may be guided by social cues, social rules, and knowledge about what the other person is doing. In order to enable users to appropriately allocate attention in situations such as the ones exemplified, attention aware systems may support group perception by supplying cues of others' activity, their level of involvement in the activity, their role, etc. Erikson and his colleagues (Erickson, Halverson, Kellogg, Laff, & Wolf, 2002) address this problem in their research on *social translucence*.

2.1.4 Presenting interruptions at the correct level of conspicuity

Notification can take a variety of forms, such as the sending of an email or of an instant message, the posting of a message in a chat box, the displaying of an item in the home page of a portal, the display of a blinking icon, or the intervention of an artificial character. The most appropriate format depends on a variety of factors, including the current state of the user (for instance the user can be busy and should not be disturbed) or the context (the user is in communication mode and consulting his/her mailbox). Whilst it has been shown that supplying information about pending tasks improves people's ability to manage interruption (C. Y. Ho, Nikolic, Waters, & Sarter, 2004), the notification modality may impact on the user activity at various levels: it may go completely unnoticed, it may smoothly integrate with the user’s current task, or it may capture the user’s attention and cause a temporary or durable focus switch. Several researchers have concentrated on the effects that different notification modalities may have on the user. Robertson and his colleagues (Robertson *et al.*, 2004) analyse two types of interruptions in debugging environment: immediate-style (i.e. interruptions that require immediate attention from the user), and negotiated-style (i.e. interruptions that the user can attend to at a chosen time). They conclude that negotiated-style interruptions are less disruptive and promote learning. McCrickard and his colleagues (McCrickard, Catrambone, Chewar, & Stasko, 2003; McCrickard & Chewar, 2003) propose to measure the effects of visual notification with respect to four parameters: (1) users’ interruption caused by the reallocation of attention from a primary task to a

notification, (2) users' reaction to a specific secondary information cue while performing a primary task, and (3) users' comprehension of information presented in secondary displays over a period of time, and (4) user satisfaction. They provide recommendations indicating, for example, that small sized in-place animation can be defined as best suited for goals of minimal attention reallocation (low interruption), immediate response (high reaction) and small knowledge gain (low comprehension). Bartram, Ware, and Calvert (2003, p. 515) propose the use of moticons (icons with motions) as an effective visual technique for information rich displays that minimise distraction. Finally, Arroyo and Selker (2003) study the effects of using different modalities for interruption in ambient displays concentrating on the effects of heat and light channels.
Scenario 16 below describes a behaviour that may be displayed by the Atgentive system by controlling notification modality (as all the other scenarios, this scenario is further analysed in section 4).

Scenario 16: Tools for various levels of interruption conspiquity
The learner must be notified about new documents available for his/her course. This information is defined as having a *low urgency* and a *high content* level. The system will pass on this information as an *email*.
Later, the learner must be notified about a real time chat meeting with the teacher that will take place in 5 minutes. This information is defined as having a *high urgency* and a *low content* level, and an *action tracking* on the "user connecting in the chat meeting". The learner is notified about the chat event by an instant message.
Later yet, if the user has not connected in the chat event, he is notified, with a further instant message, about the number of participants already in the chat meeting.

2.2 Supporting attention at the deliberative level

Whilst at the perceptual level attention is influenced by external stimuli, at the deliberative level attention is influenced by one's goals, motivations, and intentions; further, these two processes (perceptive and deliberative) constantly interact to determine one's attentional state. For example, although an external stimulus may effectively attract a learner's attention, a lack of motivation for the proposed focus will quickly divert the learner attention to another item. On the other hand, a learner may be motivated to focus on a certain item (because, for example, he/she is pursuing a certain learning goal) but an inappropriate presentation of the learning content (at the perceptual level) may prevent him/her from [easily] establishing the desired focus. This section analyses how attention may be supported at the deliberative level. In many ways, this type of support is similar to the support given to students by an experienced tutor who, on the basis of the observation of the learner, can guess (with a variable degree of certainty) how attention is being allocated and decide whether to intervene to support the current focus or propose a new one. It considers situations in which the learner:

- Looses motivation and/or stops actively pursuing a worthwhile focus; (2.2.1)
- Looses track of the planned sequence of activity and/or experiences difficulties in prioritising his/her activity; (2.2.2)
- Does not make best use of time resources or looses track of time (2.2.3)
- Is unable to select the most effective focus for his/her goal (2.2.4)
- Is unable to select the most effective focus for the goal of the group (2.2.5)

2.2.1 <u>Supporting motivation</u>

Many experiments as well as folk wisdom tell us that loss of motivation, together with tiredness, is one of the main reasons for loosing focus of attention. A system aware of the learners' attentional state can intervene to stimulate or motivate them (Angehrn, 1993). Motivations may be provided in many different forms. In certain situations it may be enough to remind the learner what he/she should be concentrating on, in most situations however, it will be necessary to supply some help in order to encourage the learner to continue the activity and, as discussed in section 1.1, interventions may take place at the regulative, cognitive, or meta-cognitive level. In a social context, motivational interventions may also take the form of the visualisation of the activity of other users. The following scenarios exemplify the Atgentive system behaviour aimed at supporting the learner's motivation.

Scenario 7: Re-attracting an idle-user attention
The student has started an activity requiring that he/she supplies some input. The student does not provide input for longer than the *maximum input inactivity time* for the task. The system evaluates whether the task being performed is still the best-suited one for the user; it verifies whether the learner is busy with offline activities. Following these evaluations the system may propose to the user: (1) to continue the task, possibly by providing motivation for the task; (2) to receive help on the task; (3) to switch to another relevant task (if available).

Scenario 7a: Re-attracting an idle-user attention (a)
The student initiates a task that he/she has never performed before. The student does not provide input for longer than the time indicated as the *maximum input inactivity time* for the task. The system proposes to the student to focus on a support task (e.g. explanation, help) for the task just initiated by the user.

Scenario 8: Re-attracting distracted user's attention
The user is active in an application that is not Atgentive enabled as a consequence Atgentive cannot assess whether the user's current focus is more "important" than any of the foci associated to Atgentive enabled applications and doesn't interrupt the user. However, being able to capture window activities such as copy and paste between windows, or frequent windows switches between an Atgentive-application and an "unknown" application, may allow the system to infer which "unknown" windows are part of the context for the current task and therefore make more informed decisions about the user activity.

The above scenarios describe situations in which users don't provide the input expected. Other situations in which learners may need motivational support are those in which learners are attending to the task but they are too slow (this is especially true with learners performing activities that are easily quantifiable in terms of the results being achieved). In this case the instructor, or in our case the system, may intervene by providing help and encouragement.

Scenario 15: Encourage slow user
The student initiates a task that he/she has never performed before. The student provides input with a frequency lower than the *minimum input frequency* for the task. The system supplies some encouragement and perhaps some simple explanations. When the learner's input frequency increases, the system gives a positive feedback.

2.2.2 Supporting task continuation and prioritization

Two problems often encountered in situations of heavy cognitive load and multitasking are related to the **correct continuation of planned activities**, and the **evaluation of relative priorities of concurrent tasks**. These problems have been studied in relation to *prospective memory* failures. Differently from *retrospective memory*, which allows us to remember facts of the past (e.g. people's names, the lesson studied yesterday), prospective memory allows us to remember planned activities in the future (e.g. go to a meeting, complete writing a paper, turning off the stove in 30 minutes) (Meacham & Leiman, 1982) and it is closely related to intentionality (Marsh, Hicks, & Bryan, 1999; Sellen, Louie, Harris, & Wilkins, 1996). Whilst prospective memory is essential for the normal functioning of our daily activity, prospective memory failures may account for up to 70% of memory failures in everyday life (Kvavilashvili, Messer, & Ebdon, 2001), they have been shown to significantly hinder performance in work and learning environments (M Czerwinski & Horvitz, 2002) and to intervene differently depending on the age of the subjects (Kvavilashvili, Messer, & Ebdon, 2001).
Prospective memory doesn't simply require remembering something, but it also requires remembering it at the *correct time*; such *correct time* may be represented by an actual time (e.g. going to a meeting at 2pm) or by the occurrence of an event (e.g. turning off the stove when the water boils). This has brought the distinction between *event-based* and *time-based* remembering tasks (Sellen, Louie, Harris, & Wilkins, 1996).
One obvious way to support prospective memory is to supply reminder services such as those illustrated in scenario 16 (section 2.1.4) where the system issues a reminder at a specified time or at the occurrence of a given event. These reminders may be particularly useful in helping users remembering

to resume tasks that have been interrupted (a study reports that in over 40% of the cases in which tasks are interrupted, they are not resumed (O'Conaill & Frohlich, 1995)).
Resuming a task, however, doesn't only require remembering to restart the task but it entails being able to somehow re-establishing the context of that task. This may require a significant cognitive effort on the side of the learner. As a consequence, whilst at deliberative level the Atgentive system may support task continuation via reminders, at the operational level, task resumption will be further supported by enabling the learner to easily re-establishing the task context (see section 2.3.1).

2.2.2.1 Supporting task continuation

Often, when resuming a task, one must be able to continue with the originally planned activity. For example, one might have started a task T1 with the plan to perform activities (i.e. subtasks):

T1A1, T1A2, …, T1Ai, T1Ai+1, …, T1An

The task is then interrupted after activity T1Ai. At the time of resumption of task T1, a significant cognitive effort may be required to recall the task continuation plan T1Ai+1, …, T1An. Providing support to task continuation at resumption time by indicating the actions to be performed next may significantly reduce the cognitive load connected to re-establish an interrupted focus.

The concept of **task** as a bundle of activities and objects that, in a sense, form the environment of an attentional focus is central to the support of attentional processes. Note that we do not refer to *task* as being necessarily result-oriented. Completing an exercise, as well as watching a movie, are considered as tasks in our framework. We also make no assumption on the level of granularity of a task. We do assume that a task **may be organised in sub-tasks** as in the brief discussion above, this organisation gives rise to a set of task hierarchies. We also assume that it is possible to define a **task context**. As also reported in (Mary Czerwinski, Horvitz, & Wilhite, 2004) we found that the definition of *task* is very much a subjective one and, in order to satisfy the need of different users, in different environments, it is necessary to maintain such definition as general as possible. For the same reason, we began our analysis with a simple definition of *task context* as including: (1) all the application windows necessary for completing the task, and (2) the task hierarchy for which the task is either the root or an internal node. Further elements to the task context may be added as needed, and they may include, as in the sample scenario 10 below, the sequence of tasks performed just prior an interruption. In general, the context for a task will be partially defined by the system (e.g. the context for an exercise contains the window describing the exercise plus the application window for completing the exercise, if it exists) and partially defined by the learner who is capable of identifying, for example, any further application window opened to complete the task.

In certain situations constraints may apply to task sequences so that the Atgentive system may simply inform the learner about such constraints.

Scenario 14: **Task sequencing**
The learner has completed a task T1 that *must be followed by* task T2. Upon completion of T1, the learner is informed that the next task to be completed is T2. Similarly, other constraints may be defined on tasks sequences, for example, that a task T1 must be completed before initiating task T2.

However, not in all situations it is possible for the system (or even the user) to know the set of subtasks (T1Ai+1, …, T1An) required to continue the resumed task. In these cases, presenting to the user information about the context of the interrupted task may facilitate resumption. A few researchers have already explored this possibility by creating systems that create logs of events that may help the user remembering about the context of an interrupted task, see (M Czerwinski & Horvitz, 2002) for an overview of this research. In the scenario below we propose that the learner might have an easier time resuming the task if he/she is reminded of the few actions performed (and documents involved) just before the interruption.

Scenario 10: **Restore historical context**
After replying to an email, and reading a document, the user is interrupted while writing a further email. When resuming this last task the system reminds the user that the last actions performed before the interruption consisted in replying to the email and reading the document.

In certain situations, learners or knowledge workers consistently perform tasks in certain sequences. If the system is capable of recognising these recurring paths in the user's actions, the information may be used to propose task continuations when tasks are started or resumed.

> **Scenario 11**: **Propose task continuation**
> After N observations the user has executed a certain task X after – or interleaved to – a task Y. The user is now focusing again on task Y, once the task is completed the system proposes to continue with task X.

2.2.2.2 Supporting task prioritisation

Multitasking not only imposes the cognitive load related to remembering what one was doing when resuming a task, but also generates the problem of keeping a clear idea of what other important tasks lay ahead. In situations of multitasking, in fact, it may be difficult to keep track of relative priorities between tasks. Several mechanisms may be used to help the learner in the allocation of attention to the most urgent task; these include the explicit definition of prerequisites or constraints for task execution, tasks deadlines, as well as interruption management. For instance an agenda can be used to help people keeping track and organizing the execution of different tasks. Such an agenda may reduce the cognitive load by giving an overall view of the way tasks are ordered, facilitating the planning and execution of tasks, allowing the user to limit the number of tasks executed concurrently, and reducing the need for the users to think about a particular task until it is necessary. On the basis of the contents of the agenda, the system may automatically send notifications about tasks due soon, relieving the users (student or knowledge worker) from remembering about these events. Task agendas may also include information about prerequisites for actions execution helping the user to allocate attention only to those actions whose prerequisites are fulfilled, or notifying the user when prerequisites for task execution become fulfilled.

> **Scenario 5: User requests notification**
> The student requests to be notified immediately and with confirmation, about any message coming from a given sender. Upon reception of the email message the system recognises that the conditions for notification are verified, consequently it notifies the user immediately (as requested). Since the user indicated that the notification is *with confirmation*, the notification is repeated at successive breakpoints until the user acknowledges it.

2.2.3 Support optimisation of time allocation

Another consequence of task complexity and/or multitasking is an increased difficulty in the selection of the most appropriate information or task to attend in the available time. For example, given a limited amount of time available to perform a task, and two pending tasks of similar urgency but different durations, if one of the two tasks can be completed within the available time and the other one cannot, it is often more profitable to attend the task that can be completed within the available time rather than the other one. These types of time-allocation evaluations are often disregarded in complex multitasking environment. This is particularly noticeable in learning environment or in stressful situations. In the former case, students may not even be able to evaluate the length of time necessary to complete a task and instructors may play an important role suggesting the best activity to be performed in the available time.

> **Scenario 2: Support to limited time resources allocation**
> The student starts reading the text for a new lecture. The system recognises that a relevant exercise task was previously interrupted (or that the exercise was previously suggested by the application). The agent also evaluates that the exercise task could be completed within the time available to the student whilst reading the text for the new lecture requires longer than the time available to the student. The system suggests working at the exercise.

This apparently simple scenario also illustrates in an eloquent manner how different support strategies may need to interact. Consider the case, for example, in which the activity that would be better suited in terms of time allocation requires resources that are not available at the time of evaluation. In this case, the "support to limited time resources allocation" strategy would have to be mediated with a "support to limited physical/people resource allocation" strategy in order to decide what the user's best focus of attention should be.

2.2.4 Support selection of most effective focus for current goal

One of the most important roles of the instructor in learning environments is that of guiding, in the appropriate situations and in the most profitable manner, the learner's attention towards the most relevant information and tasks. As a good instructor, attention aware systems could support learners by guiding them in a flexible manner so that the sequencing of activities is not rigid but it adapts to the learner current activity. For instance, in certain situations it is better to allow learners to complete a line of actions so that they themselves come to the conclusion that that action is not a profitable one, in other situations it is important to intervene to avoid frustration and de-motivation. In general, suggestions for new lines of actions are better received if they are not perceived as interrupting an activity but rather as solutions to problems that the learner has already identified.

The two scenarios below assume that some part of the system (which we call the "application") is capable of reasoning about, and evaluating the effectiveness of the user's focus with respect to some learning goal.

Scenario 4: Learning guidance
The user is reading some information and the application evaluates that the user should also read another document that he/she has not yet explored. The system evaluates the best manner to propose the new focus (on the basis of the user's current and past activity) and makes the suggestion to the user. The user disregards this suggestion (without dismissing it). The system saves the proposed focus to be able to propose it later.

Scenario 6: I don't want to do this … bug me no more!
The system proposes to perform a certain task; the user dismisses the proposal. The system will not propose the task again unless the application requires it one more time, in which case the task will be proposed the intervention with further motivation. May ask for reasons for dismissal to the user (e.g. obsolete, too busy, etc.)

2.2.5 Support selection of most effective focus in group work

In collaborative learning environments, the role of the instructor is often that of a moderator capable of identifying information and tasks that are relevant and beneficial for the whole group. In attention aware systems this can be reflected in at least three different types of actions.
First, if the system is aware about **task dependencies** between tasks [to be] performed by different group members, it may be able to suggest to a group member to attend a certain task because that task is critical for the action of other group members.
Second, by tracking access to, and actions on documents, the system may be able to inform group members about those **tasks that are receiving the most attention** from other group members. Similarly, the most popular action sequences may be used by the system to **recommend task continuation strategies**. This type of system behaviour is similar to the behaviour of collaborative recommender systems (a recent review of recommender systems can be found in (Adomavicius & Tuzhilin, 2005)).

Scenario 12: Suggest community relevant resources
As the learner accesses an online resource, say R1, the system offers a set of "related resources". These related resources correspond to those most frequently selected, by all users, immediately both before and after R1. While the user may select one of the proffered related resources, no action need be taken by the user if they so choose.
When a resource is reopened, (i.e. after the first time for that user) the user will be offered the related resources, as described above, AND any related resource accepted previously.

Scenario 13: Suggest community relevant tasks
If a sequence of N events E1 … En generated by this user matches (the event is the same and the task is the same) the beginning of a sequence of M (M>N) events of other users B1, …, Bn, Bn+1, …, Bm, then the task contained in the N+1 event of the sequence (Bn+1) is proposed to this user.

Third, in collaborative, multi-user environments it may happen that delaying, or minimising the impact of a notification message in order to optimise the performance of one user, results in sub-optimal performance for the group as a whole (as in the case in which the activity of other members of the group depends on the prompt notification and consequent response). In these situations strategies optimising individual attention allocation may not be sufficient and the selection of the **notification strategy must take into account also the state of other users**. Although most of the work on the evaluation of the cost/benefits of interruptions has been done taking the point of view of the user being interrupted, some analysis takes into account also the cost/benefit to the interrupter, and the joint cost/benefit (Hudson, Christensen, Kellogg, & Erickson, 2002; O'Conaill & Frohlich, 1995).

2.3 Supporting attention at the operational level

As indicated previously, people are very ineffective at working on too many things at the same time, because of the limited human multitasking capabilities (people can think only of one thing at a time, and as indicated by Rubinstein, Meyer, and Evans (2001), switching from one task to another is costly). People are also subject to burnout (Maslach, Schaufeli, & Leiter, 2001) when they are the objects of excessive overload and when they feel they are loosing control of the situation, a typical situation happening in a context of information overload. Burnout is characterised by exhaustion, cynicism, and inefficacy, and leads to the collapse of the performance of the individual in a work situation.

The operational support of attention consists in providing mechanisms helping the learning or working processes in a way that is effectively related to the allocation of attention in the context of many tasks to accomplish or at the many interruptions to deal with. In particular, this support should help the user to reduce the cognitive effort required to accomplish different tasks.

This section analyses how attention may be supported at the operational level. In many ways, this type of support is similar to the support we obtain by writing ourselves little notes to remind us what to do; or by hiring a secretary who acts as a filter for interruptions, helps us remember about engagements, people, or important information, and takes over the execution of certain tasks. We consider situations in which the learner:

- Needs to resume an interrupted task; (section 2.3.1)
- Deals with frequent interruption at various degrees of urgency; (section 2.3.2)
- Delegates task execution (section 2.2.3)

2.3.1 <u>Support task resumption</u>

In situations characterised by frequent interruptions or tasks alternation, a significant increase in cognitive load is related to the actions necessary to restoring the context of an interrupted task at resumption time. A diary study tracking the activity of information workers over a week reports that participants in the study rated as significantly more difficult to switch to those tasks that required "returning to" after an interruption, that "the *returned-to* tasks were over twice as long as those tasks described as more routine, shorter-term projects", and that "*returned-to* tasks required significantly more documents, on average, than other tasks" (Mary Czerwinski, Horvitz, & Wilhite, 2004, 178 - 179). A system capable of saving the context of interrupted tasks and restoring this context on demand may significantly reduce such load and minimise the resumption time. As noted above (see section 2.2.2.1) one difficulty for the designer of systems capable of this type of support is the definition of what constitutes the *task context* with respect to task resumption. Task resumption is particularly critical in the context of current desktop interfaces because such interfaces force an "application oriented" rather than "task oriented" approach to computer based activities. In order to complete a task (say write a report) the user is forced to fragment the task in subtasks (such as collecting data from a word processor to write some text, collecting data from a spreadsheet in order to paste it in the text). This artificial fragmentation of the original task imposes an increased cognitive load on the user. The scenarios below describe the behaviour of a system capable of supporting the re-establishment of task context for interrupted tasks.

Scenario 1: Support to task resumption, restoring task context (I)
The student is working at an assignment. In order to perform this activity he/she has opened the Web page of the course containing the text of the assignment (window 1), a word processor where he/she is typing some text (window 2), as well as a pdf document containing some notes from the professor (window 3). Before completing the assignment, the student switches to another task. Later the student returns to the assignment task; as soon as the student resumes the interrupted task the system proposes to restore the context of the assignment task, as it was left at interruption time, by reopening (or bring to front) the three windows 1, 2, and 3.

Scenario 9: **Support to task resumption, restoring task context (II)**
While browsing a document A, the learner has opened several windows; he/she accesses a new document B; the system proposes to the user to select the windows associated to the interrupted browsing activity on document A, in order to save the context of this activity. Later the user re-accesses document A, the system verifies whether all the windows in the context are already open. If not, it proposes to restore (one of) the saved environment(s) associated to the task of reading document A. The intervention modality will depend, amongst others, on how long the task has been idle.

2.3.2 Manage interruption: context, timing, and content

As multi-tasking and interruptions have become the norm in modern working environments (Mary Czerwinski, Horvitz, & Wilhite, 2004; Gonzalez & Mark, 2004; Mark, Gonzalez, & Harris, 2005), an obvious manner for attention aware systems to support attentional processes is to supply personalised and adaptable notification systems that reduce the disruption provoked by digital interruptions. Notification systems have been studied in a wide variety of application domains including messaging systems (Cutrell, Czerwinski, & Horvitz, 2001; M. Czerwinski, Cutrell, & Horvitz, 2000; Horvitz, Kadie, Paek, & Hovel, 2003), alerting in military operations (Obermayer & Nugent, 2000), shared document annotation (Brush, Bargeron, Gupta, & Grudin, 2001), and end-user programming (Robertson *et al.*, 2004). Interruption and notification management must take into account many factors (see for example the taxonomy in (Gievska, Lindeman, & Sibert, 2005)) that span across the various levels of support for attention (from perception, to meta-cognitive) and collectively contribute to making an interruption more or less appropriate or disruptive. Research on interruption management has covered many of these aspects which include: the context of interruption, the timing of the interruption, and its content.

Scenario 3: Notification of external events
The user is performing a task. An email addressed to the user (or other notification event), is received. The system recognises that the message is of average importance (e.g. the sender is listed in the user social network, and the subject is relevant to one of the interrupted tasks) however the system also recognises that the current task is urgent and it requires a heavy workload. The system decides to delay notifying the user about the message until the occurrence of a breakpoint in the task execution (e.g. the user completes the current activity, or starts a new activity).

2.3.2.1 Context of the interruption

Interruptions bring to one's attention events or information that may have different degrees of utility and may provoke more or less disruption in the current activity. Whilst it has been argued that in certain situations (simple primary tasks) interruptions may facilitate task performance (Speier, Vessey, & Valacich, 2003), in the more general case interruptions may generate stress (Bailey, Konstan, & Carlis, 2001; Zijlstra, Roe, Leonova, & Krediet, 1999) and hinder the performance of the primary task (Franke, Daniels, & McFarlane, 2002; McFarlane & Latorella, 2002; Nagata, 2003; Speier, Vessey, & Valacich, 2003). The cognitive load of the task being interrupted, as well as the level of involvement of the user with this task are deciding factors for the effect that an interruption may have (Fogarty, Ko, Aung, Golden, Tang, & Hudson, 2005). In general, the effects of interruption will depend on how well the new information is integrated in the context of the current activity. For example, Carroll and his colleagues (Carroll, Neale, Isenhour, Rosson, & McCrickard, 2003), focussing on the support of collaborative activities, argue that awareness information should be related to the user's current activity. Deciding what exactly the user is doing (i.e. what the current task is) is one of the most crucial aspects in interruption management. However, in current computer supported collaborative environments not only it is difficult to exactly evaluate what people are doing, but even establishing whether they are at all available may be a challenge (Fogarty, Lai, & Christensen, 2004; Horvitz, Koch, Kadie, & Jacobs, 2002). In face to face situations, human being are quite capable, in a very small time, and with a limited knowledge of people's activity, of deciding whether an interruption would be acceptable or not. Studies that have tried to replicate this human ability to evaluate *interruptibility* include sensor based predictive statistical models of interruptibility (Fogarty *et al.*, 2005; Fogarty, Hudson, & Lai, 2004) and methods for learning models from data that can be used to compute the expected cost of interruption for a user (Horvitz & Apacible, 2003; Horvitz, Koch, & Apacible, 2004).

2.3.2.2 Timing of interruption

The exact point in time when the interruption is delivered may make a significant difference on whether and how the interruption is perceived and on how much disruption it will bring to the current task. One of the most influential works in notification timing proposes four design solutions to coordinate user interruptions: ‘‘immediate, negotiated, mediated, and scheduled. Interruptions can be delivered at the soonest possible moment (immediate), or support can be given for the person to explicitly control when they will handle the interruption (negotiation). Another solution has an autonomous broker dynamically decide when best to interrupt the user (mediated), or to always hold all interruptions and deliver them at a prearranged time (scheduled)’’ (McFarlane & Latorella, 2002, p.5) and conclude that in most situations negotiation is the best choice. More recent work however has aimed at a much finer grained analysis of interruption time. In particular, several authors propose that interruptions taking place at "break points" in the user activity (e.g. when the user has finished a task and is about to start another one) are less disruptive (Bailey & Konstan, 2006; Iqbal, Adamczyk, Zheng, & Bailey, 2005). Adamczyk, and Bailey (Adamczyk & Bailey, 2004, 2005; Bailey, Adamczyk, Chang, & Chilson, 2006) propose task models that would allow for such finer-grained temporal reasoning.

Appropriate selection of interruption time is particularly critical in wireless devices because the user may be carrying/wearing such devices in a wide variety of situations. Ho and Intille propose a context-aware mobile computing device that "automatically detects postural and ambulatory activity transitions in real time using wireless accelerometers This device was used to experimentally measure the receptivity to interruptions delivered at activity transitions relative to those delivered at random times" (J. Ho & Intille, 2005, p.909).

2.3.2.3 Contents of notification

The content presented to users with a notification mechanism may range from a notification of information availability (e.g. a flashing icon indicating the presence of email) to complex awareness mechanisms (e.g. awareness display in a distributed collaborative system), to a complete switch of context (e.g. opening of a new window with a new application).

Whilst notification modality has been often studied, few authors have directly addressed the problem of the adaptation of the message content to the attentional state of the user. An example of a system that addresses this problem is READY. READY is a natural language interface that dynamically adapts to the user’s time pressure, and working memory limitations. Two prototypes have been developed: one supplying instructions for car repair (Jameson, Schafer, Weis, Berthold, & Weyrath, 1999), and one for making a phone call in an airport (Bohnenberger, Brandherm, Grossmann-Hutter, Heckmann, & Wittig, 2002). These prototypes serve to explore methodologies for assessing users’ resource limitation on the basis of their speech, and consequently bundling instructions in appropriately long sequences. Dynamic Bayesian networks and influence diagrams are used ‘‘for modelling the user’s resource limitations and making decisions about the system’s behaviour’’ (Jameson, Schafer, Weis, Berthold, & Weyrath, 1999) (, p. 81). Although, as the authors indicate, these prototypes are still experimental, they are a good demonstration of how notification content may be adapted to the user's attentional state.

2.3.3 Task delegation

Systems supporting task delegation reduce the cognitive effort required of the learner by reducing the complexity and the steps necessary to accomplish a task. In section 2.2.2 we have already discussed how task continuation may represent a problem in situations of frequent interruption. In that section it was proposed that the system may support users by reminding them the continuation plan for a task at resumption time. A further support strategy consists in the [partial] delegation of the action in the plan to the system itself.

Scenario 17: Task delegation
In a virtual learning community, the community organizer creates a message to be sent to the community, he/she can also indicate presentation style and media, the time of delivery, as well as the operations that should take place after delivery (for instance the message may be archived after it has been read by all recipients, or a reminder may be sent to recipients who did not reply). The system will take charge of completing *after delivery* actions.

In an e-learning context, mechanisms such as the ones exemplified in the scenario above may help an instructor to easily provide learning material to the students, and to ensure that the students use it effectively.

2.4 Supporting attention at the meta-cognitive level

Finally individual's and group's attention may be supported by fostering a better understanding of the way attention is managed. Support at this level consists in the provision of mechanisms helping users in observing their current attention related practices.

2.4.1 Visualize attention allocation measures

A first mechanism that can be used consists in displaying a statistical visualization of how the users are allocating their attention. For instance users may be presented with a graphical representation including information such as the different activities in which users are involved, the time allocated to each activity, the distribution of the user's effort over time (e.g. are users allocating long periods or short periods of time?), etc. Another graph may visualize statistics reflecting the number and nature of interruptions. Finally, other tools may help to visualize particular behavioural practices, such as the time between the reception of a message, and the processing of this message by the user.
Some of this information may be private for the individual user, other may be presented to the community at large.

2.4.2 Attention diagnostic

A second series of mechanisms consists in diagnostic tools helping to assess the learning of working practice of the user, and in particular to measure the level of effectiveness. One of the simplest mechanisms may consist simply in the comparison of the user's practices with others. More sophisticated mechanisms may consist in more intelligent diagnostic tools trying to discover patterns of behaviour and interpret them. Suggestion tools may provide guidance about how to improve a current attention-management practice. For instance an artificial agent (or a real person) may intervene to suggest to the user to change a practice that is not very effective for this user or for others. For example, such an agent may suggest to the user to avoid working on too many things at the same time, or to avoid continuously switching from a task to another. Another agent may suggest to the user to abandon a practice that is know to be very annoying to others and that is too disruptive (such as sending emails to too may persons, or tagging too frequently a message as high priority).

2.4.3 Learning agendas

Other mechanisms that can be used to provide meta-cognitive support consist in the implementation of learning or working agendas, in which the user can explicitly specify his/her learning objectives, and receive assistance about how to achieve them. This assistance can consist in a tool helping learners to assess the effort and the means to employ to achieve a particular objective, and later to help monitoring the progresses and identify drops of attention.

2.5 Summary of attention support at four levels

Following the discussion in this chapter, table 1 below, summarises our classification of support to attentional processes in four levels (sections *perception*, *deliberation*, *operation*, and *meta-cognition* of the table). For each level we indicate the objectives that may be achieved at that level (column 1) and for each objective we specify how that objective may be achieved (column 2), along with some sample scenarios (column 3). The scenarios with a name prefixed by the letter O refer to the use cases described in the study of the AtgentSchool application presented in Appendix 2 and they are not further detailed in this document.

Objectives	**Means**	**Scenarios**
PERCEPTION Attention support at the perceptual level: for example, facilitating access to relevant information or presenting interruptions at the appropriate level of conspicuity.		
Facilitate access to relevant information	• Preattentive features • Filtering (time and content)	
Facilitate information comprehension	• Meta-level information / Abstract • Metaphors • Connections to other info. E.g. relation to other public, community, or individual information (e.g. online, on the platform, on individual information space)	• Signaling importance or freshness of information using icons or color • Ordering information to represent temporal aspects • Summary of info for docs, , etc.
Enhance shared perception and sense of belonging	• Community level information (meta) • Information on participants in community (e.g. pictures)	• Who has read a doc. • Info about role, expertise, and activity of people
Present interruption at the correct level of conspicuity	• Based on an evaluation of interruption relevance and urgency, present interruption as more or less visible	16 – chat meeting
DELIBERATION Attention support at the deliberative level: for example, supplying tools for the control of task priorities, or by motivating users who are loosing focus.		
Support motivation	• If the user is discouraged provide encouragement (appreciation of what done so far) • If the user is bored provide stimulation (e.g. are other users proceeding well?)	7 and 7a re-attracting idle user attention 15 – encourage slow user O1-O4, O8, O16-O18, O19-O23, O28, O37-O39
Supporting task continuation and prioritization	• Agendas, Task deadlines, Notification • When a task is started the user/application can specify what should happen next and under which conditions. The user may be notified when conditions are verified	5 – notification request 11- Propose task continuation 14 – Task sequencing 10 – restore historical context O9, O11, O29, O31
Support optimization of time allocation	• Help to allocate attention in the most time effective manner	2- limited time allocation O6, O14, O25, O35
Support selection of most effective focus for current goal	• Inform users of alternative or better suited foci to achieve established goal	4 – learning guidance 6- task dismissal
Support selection of most effective focus in group work	• Inform users of community relevant tasks	12 – suggest community relevant tasks 13 - suggest community relevant tasks at Atgentive level
OPERATION Attention support at operational level: users may be supported by simplifying some attention related operations such as restoring the context of interrupted tasks, or by filtering incoming messages.		
Support task resumption	• Restore task operational and historical context	1 – task resumption 9 – restoring context (II)
Manage interruption: filtering and timing	Evaluate interruptibility on the basis of: Urgency of current task, Current task workload, User involvement in current task, User's interruption preferences. Evaluate interruption relevance: Interruption relation to current task, User declared relevance Evaluate urgency of interruption: Deadline of task in interruption, Suspended tasks depending on this task	3 – notification of external events O12, O32
META-COGNITIVE Attention support at the meta-cognitive level: the users can be supported in their reflection about how they allocate attention by, for example, providing self-diagnostic tools.		
Visualize attention allocation measures	• Time spent on each activity • Frequency of interruptions • Consecutive time allocated to activities • What are interruptions related to (e.g. email, phone) • Average response time to emails coming from selected senders, or about selected subjects • How many tasks is the user multi-tasking on?	
Diagnostic	• E.g. how user allocates attention as compared to others	
Learning agendas	• E.g. Ontdeknet learning lines	

Table 1 – Attention support at four different levels

3 Elements of the conceptual framework

This section discusses the conceptual framework for the integration of attentional support, as described in the previous section.
The essential elements of this conceptual framework are the user(s), the applications (AtgentNet, AtgentSchool), the environment, and the elements of the system for the support of attention processes, which we will call the Atgentive agents.

Figure 4 describes the main interaction paths amongst the various components of the conceptual framework.
We assume that users' attentional choices, preferences, and possible future foci, are revealed by *events* that can be captured and analysed by Atgentive agents. This analysis results in agent's *interventions*. Note that in figure 4 (and figure 5 below) agents' interventions are reported directly to the user, however, the reader should keep in mind that these pictures depict a conceptual framework rather than a system architecture; conceptually interventions will eventually reach the user (or an user interface) although, the system design may later require further modules.

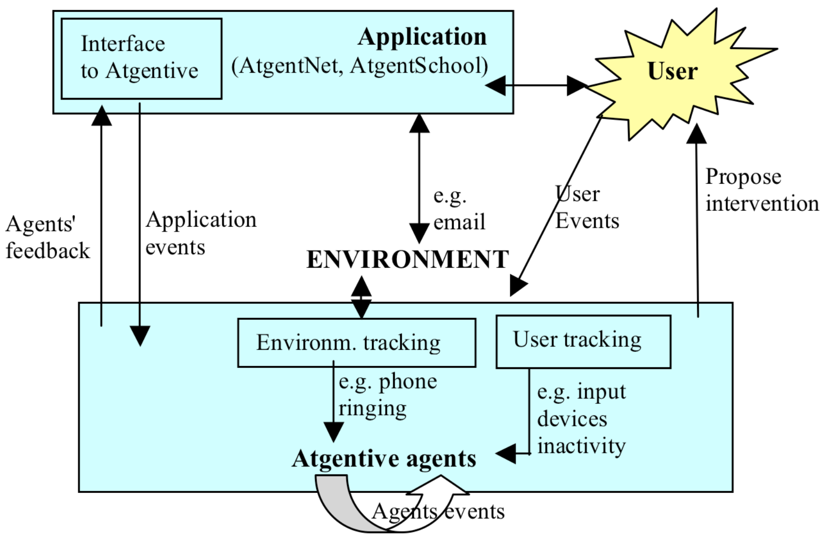


Figure 4 – Interaction paths amongst elements with examples of interactions

A more detailed conceptual image of the components of the system, which was generated by the analysis of the scenarios introduced in the previous chapter, is presented in figure 5. In this image several components are made explicit and they are described below.

3.1 Atgentive agents

What we call Atgentive agents represents the part of the system that is capable of reasoning about attention in an application-independent manner. By tracking events generated by the application and the user, it provides attention-oriented services to the applications.
Our conceptual analysis of this component is based on a multiagents system approach. Although this choice is only a tentative preliminary approach to describe and analyse the Atgentive module, we believe that it has several advantages:

1. The interventions originating from the Atgentive module will necessarily be the result of a mediation between several attention-management strategies, some of which are implicit in the scenarios we present, such as "draw attention to things that the user can (knows how to) do", "draw attention to things that the user has time to do", "draw attention to things the user may enjoy more doing", "draw attention to things that are more urgent", "draw attention to things that the user can do at this time (e.g. simpler tasks when the user is tired)", "draw attention to things that are critical for the community", and many others. The multiagent approach allows us to analyse these strategies individually (as if each agent implemented one of these strategies and then mediated with other agents to have its suggestion brought forward.

2. It is possible to analyse what the behaviour of the system would be if only certain strategies, i.e. only certain agents, were implemented; for example analysing which scenarios would be supported if only a sub-set of the strategies were implemented. This significantly reduces complexity and allows us to add new strategies as we discover them, and as more monitoring devices are added to the system, etc.
3. Most of the times only a small subset of the parameters, assumed by the sample scenarios and needed for the reasoning, will be available; other times such parameters will only be very approximate; a multi-agent conceptual model can take this into account and associate, for example, a level of certainty to various strategies (i.e. agents' proposals) depending on the available parameters.
4. Conceptually, strategies may be applied, i.e. Agents could be "fired", only when a minimal set of required parameters are available (e.g. if we do not know how much time the user has available before a significant interruption, the agent implementing the " draw attention to things that the user has time to do" will not be fired).
5. Conceptually, each strategy - i.e. agent - may be adapted to the user by tracking and updating specific fields in the user model
6. Each agent may be considered as an independent system (e.g. if one only plugs in the "draw attention to things the user may enjoy more doing")

It is assumed that Atgentive agents have tracking devices capable of observing the user; in particular, they can detect events related to the user's attentional state. User observations may span from simple detection of the usage of input devices, to more complex psycho-physiological measures such as the ones described in section 6 of the Atgentive deliverable D1.2 (Roda, 2006).

3.2 User Model

In order to provide personalised services, the Atgentive agents, and possibly the applications, will need to keep a detailed user model. Some of the components of the **user model** are also represented in the image and include:

- A list of alternative foci – these are the foci that have been suspended by the user (e.g. foci related to interrupted task) or that have been evaluated as relevant but have not yet been considered by the user (e.g. a new important email, a task that has been evaluated as relevant by the application)
- User preferences – A set of declared preferences including: maximum frequency of interruption, no-interruption time, notification modalities, tasks that shouldn't be interrupted, etc.
- Notification requests – these are events for which the user has requested notification (e.g. incoming emails; another user terminating a given task)
- Feedback log – collects the user reactions to the agents' suggestions. This log may be used to tune the agents' proposals to the specific users, and to avoid to repeatedly making the same suggestion (see also section 4.2.3)
- User specific information about various tasks, such as the task priority, and the task urgency
- A possible declaration of the length of time during which the user will be available in the current session
- Information about the user social network

The user's social network is often difficult to define. We consider the possibility of automatically detecting workgroups. A traditional way of defining workgroups is to maintain formal lists, i.e. workgroup name / users that belong to the group. An alternative would be to infer the existence of groups based on users' activities as follows:

It is logical to assume that members of a workgroup will be both working on / with the same documents and communicating with each other. Applications may track these activities. Workgroups could be identified from a combination of the following trackable events:

- Co-authorship of documents (i.e. the user who creates a document plus all users who subsequently edit that document
- Respondents (i.e. all users who contribute to a threaded conversation (add “sub-documents”))
- Mutual interest (i.e. all users who read the same document)
- Meetings (i.e. attendees of on-line meetings)
- Chats (i.e. users that chat ‘together’ (at the same time))

Two points about this methodology:

1. Other users would not be simply part of a user's workgroup or not. Instead, a quantitative measure would be available, with some users being more a part of the same workgroup than others (one user may be co-author of multiple documents while another may have replied to a few posted documents, with yet another reading a few of the same documents).
2. By searching for patterns of interaction, it may be possible to define several groups to which any one user belongs. For example, user A may have responded to several documents posted by user B under one sub-heading, but read the same documents as user C under a different sub-heading.

It would therefore be possible to implement this method of automatically defining workgroups at several levels of complexity.

3.3 Atgentive applications

It is assumed that the existing applications that will use the services of the Atgentive agents will be augmented with an Interface to Atgentive in order to communicate with the agents . , and an attention management component. Such component has been added with the aim of representing those attention-oriented enhancements that will be local to the application and will not be mediated by the Atgentive module. These enhancements may include the use of certain visualization techniques as well as more complex functionalities acting at various levels of the attention support. An example of such support is the instantiation of Scenario 10 for the AtgentNet application below.

Scenario 10: **Restore historical context**
Applied to: AtgentNet
The system will keep track of the sequence in which the user opens KAs (Knowledge Assests). For every KA, a 'list' will be held of the KAs that were selected immediately both before and afterwards (I will refer to each of these as a "contextual Knowledge Asset"—cKA).
When a user selects a KA the system will look at the last time they opened the same KA and offer the user the n (number to be determined) cKAs which (s)he had previously selected immediately before and after the original KA.
Note that for this scenario a conversation in the Chat window will count as a cKA and the contents may be displayed in a new window as if it were a 'normal' cKA document. (This is because the user may have discussed the current KA with others when it was last in use).
To reduce the cost of interruption, the user will be offered the additional documents (cKAs) only immediately upon selection of a KA. While the user may select one of the proffered cKAs (which will each open in an additional new window), no action need be taken by the user if they so choose.
Once a KA has been selected n times without accepting the contextual KAs the agent will stop offering cKAs for that particular KA (but the user may ask for contextual KAs at any time)..

3.4 Embodied Agents

The embodied agents component reflects the idea that both the Atgentive agents, and the application will, directly or indirectly, control the behaviour of the embodied agents. These agents, if requested to do so, will present information to the user. Such requests may be generated by the application (as it is already the case for the OntdekNet application that will result in the AtgentiveSchool pilot) or by the Atgentive agents.

3.5 User Interface

The user interface component has been inserted as a mediator of the communication with the user of both the applications and the Atgentive agents. Although such component may not be present at the design level, conceptually, its existence allow us to separate the "reasoning about presentation" and the actual implementation of such presentation.

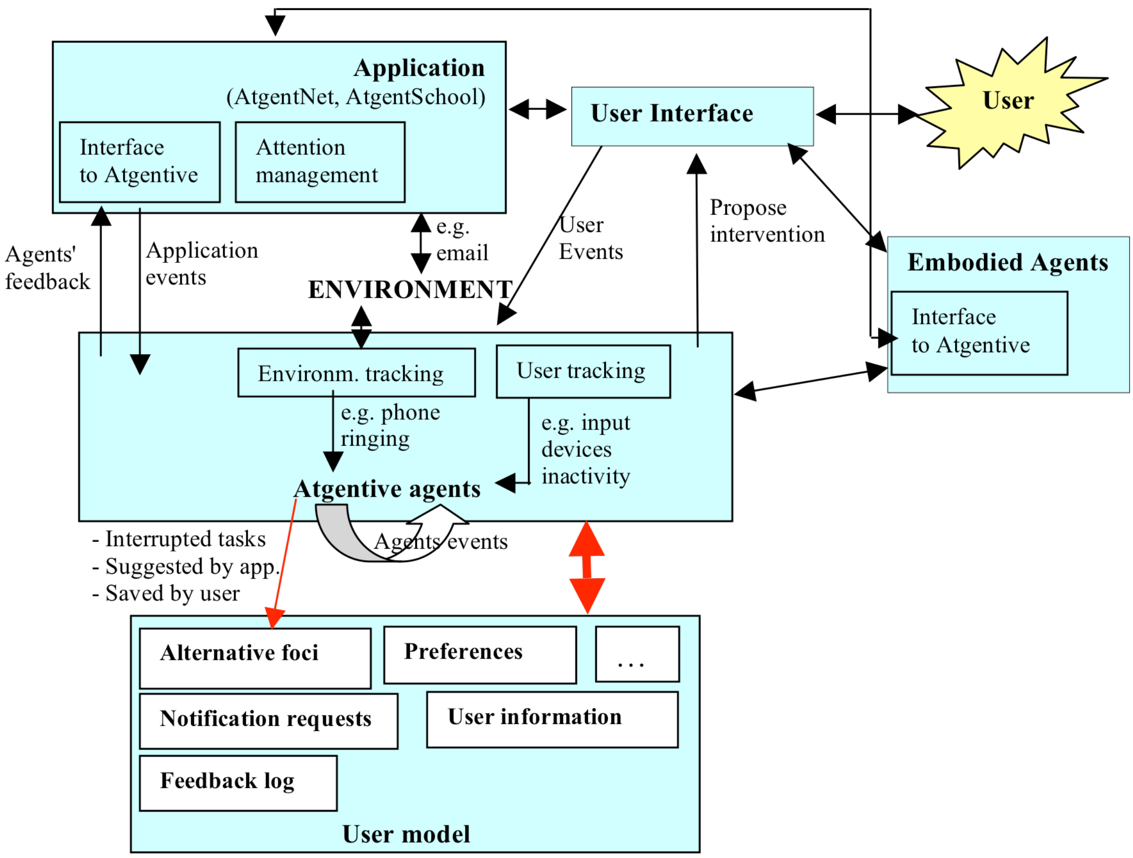


Figure 5 – The Atgentive conceptual framework

4 An event based conceptual framework for attention support

Atgentive agents are triggered by three types of events those generated by the application, those generated by the user, and those generated by the agent's own tracking mechanisms (of the user and the environment). This section describes these three types of events; the description is supported by examples of how, the scenarios presented in section 2 may be managed within the context of this conceptual model.

4.1 Events generated by the application

The application (AtgentSchool or AtgentNet) may generate events by:

1. observing the user activity: *User – Application* events,
2. observing the environment: *Environment – Application* events,
3. performing some deductive reasoning on the user's activity: *Application-suggestion* events.

These events are described below, and summarised, at the end of this section, in table 2.

4.1.1 User-Application Events

The application generates events by directly observing the user activity and recognising user's actions indicating a certain focus. User-application events report the user's activity in the application environment. The main assumption underlying the generation of these events is that the application is capable of recognising (up to a given level of granularity) the task being performed by the user. As discussed earlier, it is also assumed that it is possible to create a hierarchical description of user's tasks.
User-application events include:

- Start events – the application signals that the user has started a new task
- Continue event – the application signals that the user is continuing with a task (normally after completing a subtask)
- Complete event – the application signals that the user has completed a task
- Resume event – the application signals that the user is resuming a task
- Init event – the application signals that the user enters the application
- Stop event - the application signals that the user leaves the application

On the onset of user-application events, the agents try to determine whether there is a more appropriate focus for the user. If this is the case, the agents decide how to best propose such alternative focus to the user. Figure 6 (a simplified version of figure 5) illustrates the event-handling mechanism for user-application events. The fundamental concept behind this mechanism is that the agent should first observe what the learner is doing in order to evaluate how complex the task is, how important it is for the learner, and how relevant it is for the learning process. On the basis of this evaluation, the agent should decide whether an alternative activity should be proposed, when, and in which manner.
The event-handling mechanism is described here as a four steps process:

1. Task switch management. If the event reports a focus switch, the context of the interrupted task is saved to allow for easy resume. More formally, if the current event is a *start event* or a *resume event*, and the previous event was not a *complete event*, then the *task context* of the last active focus (i.e. the previous task) is saved.
2. Determine whether other foci would be more appropriate. This evaluation is based on the relative importance of what the learner is doing with respect to alternative foci, and on how complex it will be for the learner to resume the current activity if interrupted. If the task is just starting the agent will also consider whether the time available to the learner is enough to complete the task. If the task resumes an interrupted task the agent will offer to help restoring the context.
3. Determine the best time for intervention on the basis of the task *urgency* and of an evaluation of how far is a person from completing the current task (*estimated time to completion*).
4. Determine the best modality for intervention on the basis of the user preferences, the current task, and the intervention type.

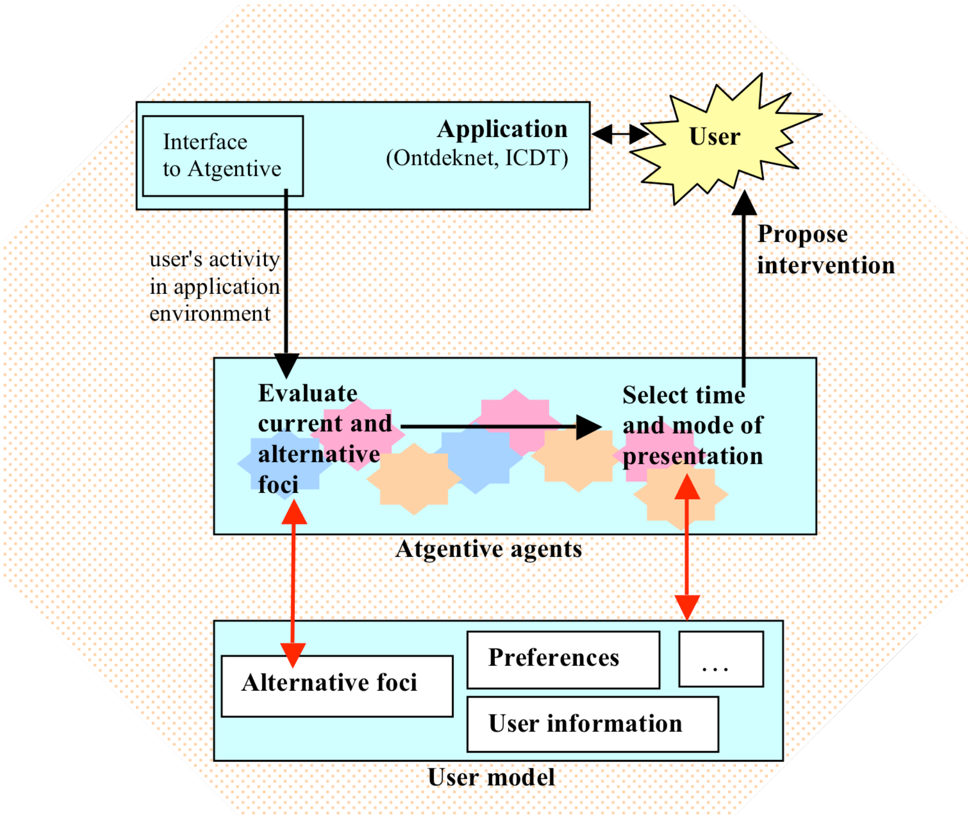


Figure 6 – The logic for handling user-application events

4.1.1.1 *Task switch management*

As proposed above, we define the *task context* as including, amongst others, the set of application windows manipulated by the user in order to complete the task. A *default task context* is statically associated to the task by the application and communicated to the Atgentive agents upon event generation. In order to have an accurate description of the context the user is consulted so that he/she can add to the default task context all other windows that the user associates to the task (note that these windows may be outside the specific application, e.g. when resuming writing this document I may need other documents currently open, some emails, etc.). We can also foresee that Atgentive agents will be able to recognise application windows related to a given tasks by recognising, for example, activities such as copy and paste between windows, and frequent shifts between windows.

Scenario 1: **Support to task resumption, restoring task context (I)**
Applied to: AtgentSchool
The student is building the mind map (current focus) using the expert's introduction diary and personal information, as well as a pdf document opened in an Acrobat window. The student switches to a questionnaire. The AtgentSchool application reports a *start* event for the new activity (questionnaire). The agent saves, possibly with the help of the user, the context of the previous focus (which includes the mind-map window, the diary and information of the expert, and the pdf document window). Later the student returns to the mind-map building activity; the application sends a *resume* event; the agent proposes to restore the saved context.

A visual demonstration of how this scenario, and scenario 9, would be managed by the system is attached in Appendix 3.
Task switch management assumes that an interrupted task will eventually be resumed or recognised as not relevant anymore. Along with *task context*, information will also be maintained about the time when the task was interrupted (*time idle*) in order to allow both (1) to estimate, upon resumption, how long the task has been idle and, (2) to eliminate from memory tasks that have not been reactivated after a given time.

Scenario 9: Restoring context II
Applied to: AtgentNet
While browsing a knowledge area A, the learner has opened several windows; the user enters a new knowledge area B (*start* event); the agent proposes to the user to select the windows associated to the interrupted browsing activity on A, in order to save the context of that activity. Later the user re-enters the knowledge area A (*start* or *resume* event), the agent verifies whether all the windows in the context are already open. If not, it proposes to restore (one of) the saved environment(s) associated to the task of browsing the knowledge area A. The intervention modality will depend, amongst others, on how long the task has been idle.

Two difficulties were highlighted with respect to scenarios 1 and 9. First of all, for both scenarios, proposing to restore the context may act as a distractor for the resumed activity. Second, in the case of scenario 1, the Atgentive system is required to have knowledge about the state of applications that are not necessarily "Atgentive enabled", for example, in order to restore windows that were open in Acrobat Reader, or Microsoft Word, Atgentive must be able to act at the Operating System level.
To address the first problem we conducted some experiments reported in (Rudman & Zajicek, 2006 submitted-a, 2006 submitted-b) and further detailed in the forthcoming deliverable D.4.2 (Rudman & Zajicek, 2006). We found that participants generally appreciated being given assistance in restarting the task, finding it helpful and being pleased that someone was trying to help. However, it seems that there was a separate group with an identifiable reason for not wanting the “assistance”. These participants, on seeing the task for a second time (or possibly in the intervening time) decided to take a different approach to solving it. Thus, when the "assistance" was made available to them it was no longer relevant. How they dealt with this varied, from disinterest to annoyance. It is possible that the offer of (irrelevant) information was taken as a suggestion that their new direction was somehow being called into question. In AtGentive, then, we need to be aware that the user may restart a task in a different manner, and offering contextual information from their previous work may not be helpful.
In order to address the second problem, we developed a simple, "proof of concept" prototype that allowed us (and the user) to group windows from various applications in a "task" bundle (see figure 7) that could be saved and restored as needed. The data-collection issues related to task resumption have been reported in (Clauzel, Roda, & Stojanov, 2006) and will be further detailed in D3.1 "Implementation of the attentive agent module (early prototype)".

The prototype consists of a global graphical interface (shown in figure 7) composed of several modules which log the user's activity while providing her/him a way to manipulate his computer environment.
The prototype allows the system to monitor the windows, applications, documents, emails, tasks, and their properties: size, content, coordinates, how often and how long they are used, etc. Instead of dealing with several windows and icons, users create several groups of graphical elements called ”tasks“ which contain all the application windows that are necessary to do a specific activity. Then by using the task management tools, they can organize their work in a global manner rather than having to handle individual application items. Users can create, delete, and switch between existing tasks (collection of windows). Tasks are presented to the user in the *context management bar*. The system collects data on the user’s high level activity whilst allowing human or artificial observers to interact in a more or less conspicuous manners, e.g. by using animations, sounds and pop-up dialogues (*intrusive* and *non-intrusive* information delivery systems). As the first results look promising, we plan to improve our prototype by tracking user’s low level actions, starting with functions activated within the Atgentive application.

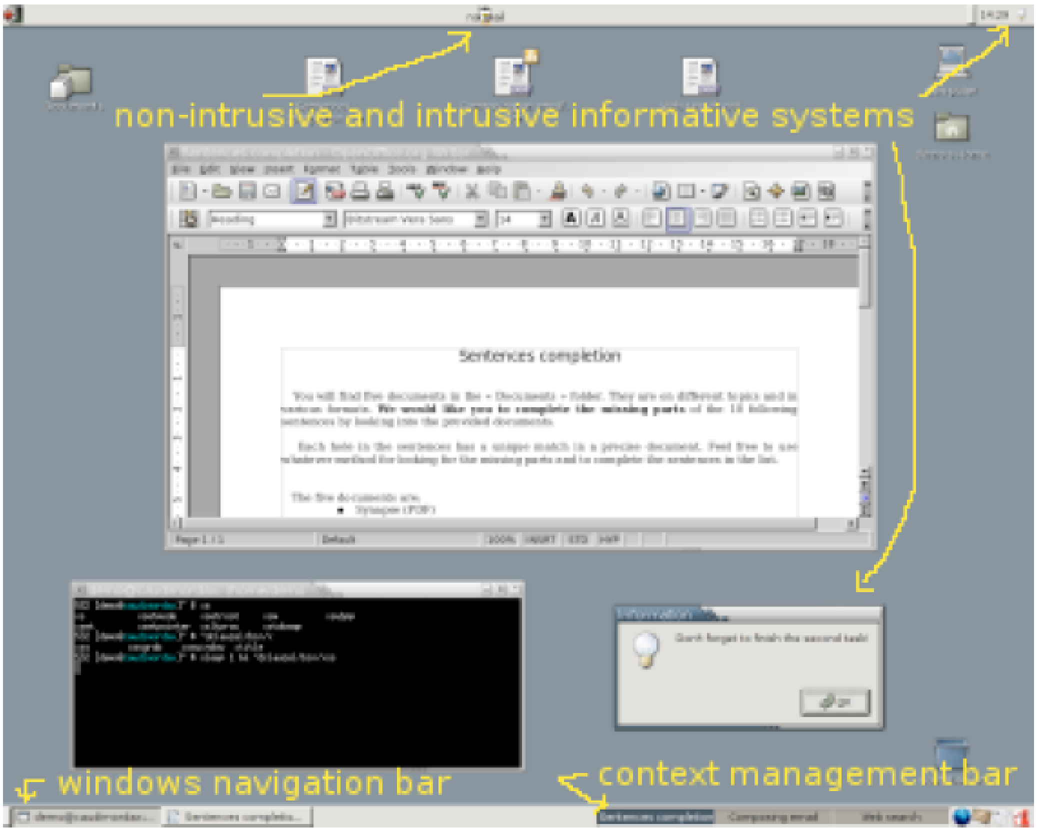


Figure 7 – A proof of concept prototype for task switch management

4.1.1.2 Determining alternative foci

As the user works the agents will keep track of possible alternative foci for the user; the list of possible alternative foci is stored in the user model (see figure 5). These foci are collected as various events take place:

- User-Application events of type *start* may signal that the user has switched focus. In this case the focus on the task that has been interrupted is saved as a possible alternative focus (along with the task contexts as described in section 4.1.1.1 above);
- Application-suggestion events of type *propose focus* (see scenarios 4 and 14 in section 4.1.3) signal that the application has evaluated that a different focus may be better for the user. If the user does not accept this suggestion the suggested task is saved as a possible alternative focus;
- User events of type *remind-me* may signal that the user wants to be reminded about a certain focus; such focus is saved as a possible alternative focus;
- User tracking events of type *foci sequence* may signal that the user has entered a recurrent sequence of foci and therefore he/she may want to continue the sequence in a standard manner (see scenario 11 in section 4.3.1)

Given the current user focus and the list of possible alternative foci, the opportunity to propose an alternative focus may be evaluated on the basis of:

- the **relative importance** of what the learner is doing with respect to what he/she could be doing (alternative foci),
- the **cognitive load involved in resuming** the current activity if this is interrupted, and
- the **time available** to the user for completing the task

Several parameters determine the **relative importance** of what a learner is doing.
One may consider how important the task is to the learner: <u>*priority*</u>. The user may directly provide the value of task priority. The agents may also evaluate task priority on the basis of user observation (psycho-psychological measures, but also others such as: how much time has the learner already allocated to the task?) and task structure (e.g. is the task associated to other high priority tasks or goals?)

Another element that may be considered in the evaluation of the importance of the learner's activity is *task relevance* with respect to the learning goal. Task relevance is strictly application dependent and it may vary with time, therefore it should be dynamically evaluated by the application.
Finally, one may consider that more urgent tasks are more important than others: *urgency*. Task urgency is normally indicated by the user, however the application may also be able to provide estimates, for example by maintaining a user agenda indicating when tasks are due.

In order to evaluate the **cognitive load involved in resuming** the current activity if this is interrupted, one could take into account an *estimated workload* associated to the task: the higher the workload the more complex it will be for the user to resume the task (Jersild, 1927; Rubinstein, Meyer, & Evans, 2001). Estimates of the workload are supplied by the application; agents can also build estimates of workload by observing the user performing similar tasks (task similarity would be indicated by the application).
Another, purely mechanical, measure that may indicate how complex resumption would be is the *size of the associated environment* (how many windows have been opened to work on the task?).

Finally, the **time available** to the user for completing the task may play a role in deciding whether a certain focus should be proposed to the user. As noted in section 2.2.3, given a limited amount of time available to perform a task, and two pending tasks of similar urgency but different durations, if one of the two tasks can be completed within the available time and the other one cannot, it is often more profitable to attend the task that can be completed within the available time rather than the other one.

Scenario 2: Support to limited time resources allocation
Applied to: AtgentSchool
The student starts working at the mind-map (start event). The agents recognise that a relevant exercise task was previously interrupted (or that the exercise was previously suggested by the application). The agent also evaluates that the exercise task could be completed within the time available to the student whilst the mind-map task requires longer than the time available to the student. The agent suggests working at the exercise.

Relative importance of tasks, and resumption complexity seem to be strong indicators of whether maintaining a given focus is preferable to switching to a new task or vice versa. Time available, however, appeared a much weaker indicator, for this reason we decided to conduct an experiment (reported in (Rudman & Zajicek, 2006 submitted-b)and further detailed in the forthcoming deliverable D4.2 (Rudman & Zajicek, 2006)) that could give us an indication on whether, interventions such as the one described in the scenario 2 above would be beneficial to the user. We found that telling participants that an alternative task was quicker than the one they were attempting was generally seen as annoying and/or unhelpful. The reason given was consistent: once users had begun a task they were committed to that course of action. They did not want to change and start a different task. Even when they believed that the advice was correct, and it would indeed be quicker at that point to change tasks as suggested, they did not necessarily take the advice. There seems to be a critical point at which the effect of the suggestion changes; it is the point at which the participant makes a decision as to which task to work on. It seems that the sooner such a suggestion is made the better. In addition, believability and related issues (trust, and likeability for example) would appear to be important attributes for the agent. Also, giving details of why the suggestion has been made may help make the intervention less annoying, as it would assist the person to justify any change of decision, something that seemed important in our study.
We are currently exploring an alterative approach. Instead of having the agents react to the user task choice by suggesting a shorter task, Atgentive could display the list of suspended foci with some annotation (e.g. a colour code) indicating whether the task can be completed within the available time. Information about the estimated time to completion for suspended tasks could also be made available to the user to facilitate the choice.

To summarise, possible foci (current and alternatives) may be evaluated on the basis of the following parameters:

- Whether they are urgent
- Whether they are related to the current task
- Whether they are related to an high priority task
- Whether they are defined by the users as being important
- Whether they are associated to a heavy workload

- Whether they can be completed within the available time

4.1.1.3 Determining the best time for intervention

If the agents decide that an intervention is necessary, i.e. they find that a "better" focus (or a set of alternative foci) should be proposed to the user, they would also evaluate what is the best time for intervention. In general, time of intervention should be evaluated by considering the trade-off between the urgency of the intervention (task-urgency of the associated task), and the need to minimise disruption to the user's activity. See also scenarios 3, 5, and 16 below.
As a general rule, one could expect the system to reproduce simple "politeness" rules by which, if the interrupting agent believes that the partner is about to complete a task (e.g. a conversation, the writing of a document, etc.), then the interrupting agent will withheld the interruption until the task is completed. If an estimate of the time necessary to complete the current task has been provided by the application, Atgentive may interrupt tasks only if the *estimated time to completion* is longer than a given *patience time* of the agent. The *patience time* indicates how long, no matter the urgency of the intervention, the agent is prepared to wait before interrupting the user. More complex calculations may involve evaluating the trade-off between task-urgency and *estimated time to completion*.
In general, depending on task urgency, at least three possible times for interruption may be considered:

1. Interrupt now
2. Interrupt at end of current task if completed by <given time>
3. Interrupt as soon as a new *user-application* or *user* event takes place
4. Interrupt only after complete event for the current task

Note that user-application and user events are considered *breakpoints*, i.e. they indicate some significant break in the user activity. Previous research (Adamczyk & Bailey, 2004; Bailey & Konstan, 2006) has demonstrated that interruptions at breakpoints are less disrupting than interruptions at any other point of the user activity.

4.1.1.4 Determining the best modality for intervention

By *intervention modality* we refer to the manner in which the agents communicate with the user.

Atgentive modalities can be categorised in three main classes:

1. Embodied agents - an embodied agent presents the intervention to the user
2. Language or cue based agents - the intervention is presented by a "conceptual" agent (e.g. an agent window, or a voice, etc.)
3. Non-agent based - users will have no hint of the existence of agents, they will simply see some text, or image, or hear a sound recommending a given action

Within these three main modalities it is possible to identify further sub-modalities related, for example, to:

1. The size on screen of the intervention
2. The media modality/ies selected (e.g. text, voice, sound, image)
3. The possible colour choice, tone of voice, loudness, text size, and many others

Modality selection should be based at least on two parameters: the characteristics of the intervention, and the user's preferences.
The characteristics of the intervention include:

- The complexity of the message that needs to be transmitted to the user (is this somehow related to the complexity of the task?)
- The urgency of the intervention
- The level of certainty of the proposed intervention (this is related to "how much better" the agents believe the proposed focus is with respect to the current one)

The user's preferences include:

- Preferences on modalities explicitly declared by the user (*set preferences* event)
- Agents' inferred preferred modalities (either by observation of the user, or as a result of a *feedback event*)

In a first instance we will study ways in which modalities are selected on the basis of a coarse specification of the desired effect. Such *coarse modality indicators* may include:

- High interruption / low interruption (how noticeable the message should be?)
- Online reach / off line reach (should the notification be noticed by an off-line user?)

4.1.2 Environment-Application Events

The application generates events by observing the environment. This assumes that the application can receive input from other sources than the user.

Environment-application events may signal activities such as the arrival of an email, or the addition of a file to a shared board (see, for example, scenario 3 below). These events (*new information available* events) are communicated by the application to the Atgentive agents who evaluate whether the event should be communicated to the user or stored as an alternative focus for later consideration. If the agents decide to intervene, they also assess the best time and modality for intervention. The event-handling mechanism can be organised in a four steps process (each step is further described below):

1. Determine whether event should be communicated to the user.
2. If not, store as alternative focus.
3. Determine the best time for intervention.
4. Determine the best modality for intervention.

Conceptually, this process differs from the process described for the management of user-application events only slightly. The main event's attributes, determining the importance of the event to the user, in this case are: the *event originator* (e.g. the sender for an email, the identity of the contributor in a bulletin board entry), and the *subject of the message* (e.g. email subject, entry subject, file description). In order to determine the importance of the event to the user, Atgentive agents can relate these two pieces of information to:

1. Possible notification requests expressed by the user (*notify me* event),
2. Relative importance of the event originator within the *user social network*. Note that this assumes that the user model will contain a description of the user's social network providing enough information for the agents to be able to evaluate, either statically or dynamically, the importance of an individual within the social network. Static evaluation may involve simple *importance* indicators associated to each element of the social network. Dynamic evaluation will require information such as association of elements of the social network to tasks or keywords, and/or explicit relations between different elements of the social network and with the user.
3. Possible relations of the message subject to high priority tasks as indicated by the *task keywords*.
4. Possible relations of the message subject to interrupted tasks as indicated by the *task keywords*.
5. Possible requests by the user of specific interruption frequency for the message type (see scenario 16 below)

The following scenarios 3 and 16 are two examples of how environment-application events may apply to the AtgentNet and AtgentSchool applications.

Scenario 3: Notification of external events
Applied to: AtgentSchool, AtgentNet
The user is performing a task (e.g. user is working at an assignment in the AtgentSchool application, the user is browsing a space in the AtgentNet application). An email addressed to the user (or other notification event), is received by the application. The application originates a *new information available* event. The agents recognise that the message is of average importance (e.g. the sender is listed in the user social network, and the subject is relevant to one of the interrupted tasks) however the agent also recognises that the current task is urgent and it requires a heavy workload. The agents decide to delay notifying the user about the message until the occurrence of a breakpoint in the task execution (marked by a new *user-application*, or *user* event).

Scenario 16: Tools for various levels of interruption conspiquity
Applied to: AtgentNet
For each entry on the platform, the AtgentNet application generates a *new information available* event indicating that this is a "new platform entry", that the urgency is *low*, and that the *content level* is *high*. The user has indicated, with a *set interruption frequency* event, that the maximum interruption frequency for the "new platform entry" information is *weekly,* and that the interruption modality should be by email. The agents collect all "new platform entry"

information and inform the user with a weekly email summarizing the activities of the last period (such as the number of messages that have been posted, the title of the messages, and some indicators of the activity of the community).
Later, the AtgentNet application generates a *new information available* event indicating that:

1) this is a "new chat meeting",
2) that the urgency is *high* (the meeting will take place in five minutes),
3) that the *content level* is *low*, and
4) that the application requires notification if the user does not connect to the chat within 5 minutes.

The agents notify the user about the chat event with an instant message.
Since the user does not login in the chat within 5 minutes, the agents notify the application.
The application generates a further *new information available* event that results in the user receiving a further instant message, about the number of participants already in the chat meeting.

4.1.3 Application-suggestion Events

The application generates events by reasoning on the user activity. This assumes that the application is capable of making inferences on the quality of the user's chosen activity and generating events notifying the Atgentive agents of alternative tasks for the user. We can consider two types of application-suggestion events: *propose focus* events and *retract focus* events. The former events inform the Atgentive agents about a better focus that the application has selected for the user. The latter events inform the Atgentive agents that a previously proposed focus proposed by the application is no longer valid.
Application-suggestion events are generated by the application in order to inform the Atgentive agents about tasks that the application deems better suited for the user at the current time (see scenario 4 below). As such, Atgentive agents will not evaluate the appropriateness of the intervention, nor its timing (which have already been determined by the application), rather they will just select the most appropriate intervention modality (see section 4.1.1.4). If the proposed new focus is not be retained by the user, it will be added to the list of alternative foci.

Scenario 4: Learning guidance
Applied to: AtgentNet
The user is visiting one of the platform's *knowledge area* and the application evaluates that the user should also visit another knowledge area, which he/she has not explored. The application generates a *propose focus* event. The agent evaluates the best manner to propose the new focus (on the basis of the current and proposed foci characteristics) and makes the suggestion to the user. The user disregards this suggestion (without dismissing it). The agents save the proposed focus to be able to propose it later.

4.1.4 Summary of Application Events

Table 2 summarises the events generated by the application.

APPLICATION EVENTS (Events generated by the application)			
Event name	**Description**	**Examples** **Comments**	**Possible parameters**
User-Application Events			
Start event	User starts a new task	Ontdek: the student enters the expert assignment phase Assumes task recognition	Default task context, estimated time to completion, maximum idle input, task keywords, task relevance, estimated task workload, task motivation, lower alertness, task help.
Continue event	User switches sub-task continuing on a super-task	Ontdek: the student browses expert information Assumes a hierarchical task description	Default task context, estimated time to completion, maximum idle input, task keywords, task relevance, estimated task workload, task motivation, lower alertness, task help.
Complete event	User has completed a task	Ontdek: student has completed the personal information sheet	none
Resume event	User resumes a task previously interrupted	Assumes task tracking on the side of the application. The recognition of this event is not essential (Atgentive agents may be able to recognise start events as resume)	Default task context, Estimated time to completion, maximum idle input, task keywords.
Init event	User enters the application		Maximum idle application
Stop event	User leaves the application		none
Environment-Application Events			
New information available event	The application recognises that the user could focus on newly available information	Arrival of an email message, new information posted on a BB.	Event type, message subject, event originator
Application-suggestion Events			
Propose focus event	The application recognises that the user could focus on a different task, or on a alternative implementation of the task.	Recognition of sub-optimal task performance.	Default task context, estimated time to completion, maximum idle input, task keywords, task relevance, estimated task workload, task motivation, lower alertness, task help. An explanation of the rationale for proposing the focus should be supplied with the event.
Retract focus event	The application communicates that a previously suggested or interrupted focus should not be resumed anymore.		

Table 2 – Events generated by the application

4.2 Events generated by the user

The user generates events by interacting directly with the agents. Users may interact with the agent in order to:

1. Supply information,
2. Request services,
3. Provide a feedback about the agents' interventions.

This assumes that the application interface is augmented with the possibility for the user to interact directly with the agent. Various types of events generated by the user are described below, and summarised, at the end of this section, in table 3.

4.2.1 User Information Supply events

Users may communicate directly with the agents in order to supply information about their attentional preferences and constraints. Users may indicate preferences such as, the maximum frequency of interruption (see scenario 16 in section 4.1.2.), or the preferred method of interruption, or a time when they don't want to be interrupted (no-interruption time), or a task that shouldn't be interrupted, etc. Users can also indicate constraints such as tasks deadlines, or how long they will be available until the next off-line interruption. Scenario 2 in section 4.1.1.2, for example, uses this latter information.
All *user information supply* events will result in the agents updating the user model, for example by modifying the maximum frequency of interruption (*set interruption frequency* event), or the preferred method of interruption (*set preferences* event), or the no-interruption time (*set preferences* event), or the time available for this online session (*set time available* event).
A subset of *user information supply* events will also trigger a re-evaluation of the most appropriate focus. Such events include the *set task deadline* event, and the *set task priority* event.

4.2.2 User Service Request events

Users may communicate directly with the agents in order to request attention-related services. Such services may include requests for notification, requests to restoring the context of a certain task, requests for help in interacting with the agent, and many others (see summary in table 3 below).

Each of the *service request* events requires an ad hoc handling by the agent.

***Notify me* events** (see scenario 5) are requests by the users to be notified of events (*tracked event*) such as the reception of an email. Typically the user will request notification about *environment-application* or *environment tracking* events. A notify me event may track an individual event, or a set of events (e.g. all the emails with a certain subject).
The agents will react to this type of events by adding the request to the *notification requests* field of the user model, and by verifying whether the event has already taken place. The latter action corresponds to verifying whether the event has already been stored in the *alternative foci* list. If this is the case an intervention will take place. *Notify me* events may allow the user to indicate whether the notification should be acknowledged or not (*notification acknowledgement*), the *time for interruption*, and the *coarse modality indicators* (see section 4.1.1.4). If the user requires an acknowledged notification, the agent will repeat (or maintain) the notification message until the user will have acknowledged it. If the user indicates a time for interruption (i.e. one of: immediate, at break, at complete) the agents will not evaluate the time for interruption but will follow the user request.
Whether the user has indicated these parameters or not, the best time for intervention will be evaluated in a manner similar to the one discussed in section 4.1.1.3 and the modality of intervention is evaluated as discussed in 4.1.1.4.

Scenario 5: User requests notification
Applied to: AtgentSchool, AtgentNet
The user requests to be notified immediately and with confirmation, about any message coming from a given sender (*notify-me*). The application, upon reception of email messages, notifies the agents (*new-focus*). The agents recognise that the user wants to be notified about the email. The agents notify the user immediately (as indicated by the *notify-me* event). Because the user indicated that the notification is *with confirmation*, the notification is repeated at successive breakpoints until the user acknowledges it.

Save the current context events are handled by saving the current task context as described in section 4.1.1.1

We assume that the list of suspended foci is always available to the user who may request to **resume a tasks** (for example by clicking on the corresponding focus name). These events are handled by restoring the context of the suspended task, which becomes the current focus. In certain situations (i.e. when the user interrupts another task in order to resume the selected one) the context of the newly interrupted task is saved.

Remind me events report the user's request to be given a *reminder message* at a given time (*reminder time*) or corresponding to a given event, or set of events (*tracked event)*. As in the case of *notify me* events, the user may supply a *time for interruption*, indicate a *notification acknowledgement*, and specify *coarse modality indicators*. The handling of these events is similar to the handling of *notify me* events, except that the user is simply given a reminder message, no switch to the task associated to the tracked event is proposed.

The handling mechanisms for **help events** will implement the contextual help for the Atgentive system. It may include, for example:

- Explanation for why a certain focus has been proposed
- Instruction on how to remove an interrupted task from the alternative foci list

4.2.3 User feedback events

Users may communicate directly with the agents in order to supply feedback on the agents' behaviour. These events allow users to explicitly accept or dismiss agents' suggestions, but also they allow users to indicate to the agents how "good" an intervention has been: was it useful? Did it come at right time? Was it too disruptive? This latter type if events may be used to tune the agents' behaviours to the particular user and it will also be important for our project evaluation.

User feedback events assume that Atgentive agents allow user to respond to their interventions by either:

- Disregard intervention (no user action …)
- Dismiss intervention (explicit dismissal of agents' proposal(s))
- Accept intervention (explicit acceptance of agents' proposal(s))
- Comment intervention (explicit comment on the agents' proposal(s))

Whilst the latter three user responses correspond to specific user events (see *user feedback* events in table 3 below), a disregarded intervention does not generate new events. In general it will be difficult to assess whether an intervention has been disregarded. A user, in fact, may undertake suggested actions with some delay without explicitly stating so. If we can recognise non-explicitly accepted interventions (as, for example, in the case of an user-application event that immediately follows the intervention and corresponds to the suggested task) these should be recorded along with explicitly accepted ones.

Dismiss intervention events amounts to asking the Atgentive agents to avoid proposing the suggested task again. The agents will therefore remove the dismissed task from the alternative foci list. The user's dismissal will also be recorded in the user's *feedback logs*. See scenario 6 below.

Scenario 6: I don't want to do this … bug me no more!
Applied to: AtgentSchool
A child has logged in the AtgentSchool application and is expected to complete the *introduction* activity. The child is new to the activity (he/she has never completed the *introduction* before), has been rated by the teacher as a weak student, has been inactive for a few minutes, and has not reached the *introduction* screen yet. The agents propose some navigational help explaining how to reach the *introduction screen* (e.g. " By clicking on the top left button you will reach the *introduction* screen"). The child dismisses the suggestion. Because the intervention has been dismissed, the Agents will not propose this type of intervention again unless the application requires it, in which case the task will be proposed with further motivation (e.g. "Before you start working at the mind map you must introduce yourself; it looks like you are having troubles reaching the correct screen. By clicking on the top left button you will reach the *introduction* screen").

Upon reception of ***accept intervention events***, the agents will perform the associated action (e.g. restore a context) and, they will record the acceptance in the user's *feedback logs*.

Comment intervention events allow users to indicate to the agents how "good" an intervention is in terms, for example, of its usefulness, timeliness, disruptiveness, or clearness. The information collected from these events will be recorded in the user's *feedback logs*.

4.2.4 Summary of User events

Table 3 gives examples of events generated by the user.

USER EVENTS (Events generated by the user)			
Event name	**Description**	**Example** / **Comments**	**Possible parameters**
User information supply events			
Set time available event	User indicates a time when he will interrupt the activity		Time available
Set task priority event	User indicates the priority that he assigns to a certain task		Task priority
Set task deadline event	User indicates a deadline for the task		Task deadline
Set interruption frequency event	User indicates the maximum frequency of interruptions		Maximum frequency of interruption; event type; interruption modality
Set preferences event	User sets preferences information	Display/hide embodied agent / This event may have to be split in several events	May include preferred modalities, location on screen of certain elements, etc.
Service request events			
Notify me event	User informs the agents about events for which he wants to receive notification	Notify me about emails from experts	Tracked event (event originator, message subject), notification acknowledgement, time for interruption, coarse modality indicator
Save current task event	User requests to save the relevant task context for future resumption of the current task	Ontdek: Save "filling personal information sheet"	none
Resume task event	User requests to set the context in order to resume a task that was previously interrupted	Ontdek: Restart "filling personal information sheet"	Task identifier (this does not have to be explicit, e.g. the user select the task in a list of interrupted tasks)
Remind me event	User requests to be reminded of something (possibly in correspondence to an event)		Reminder message, reminder time or tracked event (event originator, message subject), notification acknowledgement, time for interruption, coarse modality indicator
Help event	User requests help on the current task	Note that this help is related to the agent behaviour and it is not application dependent	Why are you suggesting this task? Can you provide more information on this task?
Feedback events			
Comment intervention event	User explicitly provides feedback to the agents on the usefulness of an intervention	Don't interrupt me again during this task. This was a good suggestion! / May be used for project evaluation	Some comment (open / close choice?)
Dismiss suggestion event	User indicates that a suggested focus should not be further suggested	The agent suggest to restore a task context the user indicates that it is obsolete	Task id (the one proposed)
Accept suggestion event	User indicated that a suggestion is accepted	Agent suggest to switch focus, the user accepts	Task id (the one proposed)
Why event	User asks the reasons for a given suggestion	Agent suggest to switch focus, the user asks why / Note: this event is only listed here but not discussed elsewhere	Suggestion motivation

Table 3 – Events generated by the user

4.3 Events generated by tracking devices

It is assumed that Atgentive agents are capable of tracking some attention-related events that are application-independent. These events are the result of either tracking the user individually, a group of users, or the environment. Various types of events generated by tracking devices are described below, and summarised, at the end of this section, in table 4.

4.3.1 User tracking

User-tracking devices generate events reporting user's states. These devices report particular user's states by either directly observing the user (psycho-physiological measures) or by tracking the user activity. In the general case Atgentive agents will use both information received from the application and information reported by the devices to recognise significant attention-related event. For example, the application may provide information about the upper limit for input inactivity, or the lower limit for user 'alertness', during a task; this information is then used by the agents to decide whether the tracked user activity or alertness are outside the admissible limits (section 6.1 deliverable D1.2 (Roda, 2006))

The Atgentive agents' user tracking component generates ***idle input events*** whenever the user has not provided input for a time longer than a *maximum idle input* time (see scenario 7). The *maximum idle input* time is one of the parameters that the application associates to user-application events; the parameter is used by the agents to set the tracking devices so that they report *idle input events* at the appropriate time for the specific task.

Low input frequency events are generated whenever the user has provides input slower that the *minimum input frequency* (see scenario 15). The *minimum input frequency* is one of the parameters that the application associates to user-application events; the parameter is used by the agents to set the tracking devices so that they report *low input frequency events* at the appropriate time for the specific task.

When *idle input,* or *low input frequency events* are generated, the agents evaluate if the task being performed is still the best suited (with a procedure similar to the one described in section 4.1.1.2).
Several considerations may enter in the decision about the best time for intervention. First, if the system is able to detect that the user is not working at the computer, the intervention may either be withheld until the return of the user or it will take a form that would be noticed by a user who is not looking at the computer (e.g. a sound more or less loud). Second, if the system is capable of detecting users' activities off-line (e.g. the user being on the phone, or speaking to a colleague, or being in a meeting) the time for intervention may be adapted to this knowledge (e.g. wait until the end of the phone call, or the meeting). Once the modality has been selected, with a procedure similar to the one described in section 4.1.1.4, the agents propose to the user to either: (1) to continue the task, possibly by providing motivation for the task; (2) to receive help on the task; (3) to switch to another relevant task (if available).

Scenario 7: Re-attracting an idle-user attention
Applied to: AtgentSchool
The student has started browsing the expert's information (*start event*). The student does not provide input (*idle input*) for longer than the time indicated as the maximum input inactivity for the task. The agents evaluate if the task being performed is still the best-suited one for the user. The agents consult the user's agenda to verify whether he/she is busy with offline activities. The agents propose to the user to either: (1) to continue the task, possibly by providing motivation for the task; (2) to receive help on the task; (3) to switch to another relevant task (if available).

Scenario 7a: Re-attracting an idle-user attention (a)
Applied to: AtgentSchool
The student works at the introduction (*start event*). He/she has never performed an introduction before. The student does not provide input (*idle input*) for longer than the time indicated as the *maximum input inactivity* for the task. The agents propose that the application should provide support for the introduction task. This support may depend, amongst others, on the input already supplied by the student.

Scenario 15: Encourage slow user
Applied to: AtgentSchool
The student starts with the *introduction* task that he/she has never performed before. In the *start* event the application has indicated a *minimum input frequency* for the task. The student provides input with a frequency lower than the *minimum input frequency* (*low input frequency event*). The system supplies some encouragement and perhaps some simple explanations relative to the *introduction* task. When the learner's input frequency increases, the system gives a positive feedback.

Atgentive agents may also track user activity with the aim of recognising significant paths in this activity. ***Foci sequence*** events are generated whenever the tracking device recognises that the user has entered an established sequence of foci. Established sequences of foci may be deduced by the agents' tracking devices as in the example proposed in Scenario 11 or they may be established by the application by specifying constraints on task sequences as exemplified in scenario 14.

Scenario 11: **Propose task continuation**
Applied to: AtgentNet
After 10 observations the user has looked at the platform's action-log immediately after reading all new messages on the platform 8 times out of 10. The user is now focusing again on the new messages, once this task is completed the agents proposes to continue the activity by looking at the platform's action-log.

Scenario 14: **Support to task continuation: required sequence**
Applied to: AtgentSchool
The application has informed Atgentive that the task *login* must be followed by the task *introduction*. Once the learner has completed a task *login* he/she is informed that the next task to be completed is the *introduction*. Similarly, other constraints may be defined on tasks sequences, for example, the *introduction* must be completed before *contacting the expert*.

4.3.2 Environment tracking

Environment-tracking devices generate events reporting environmental changes that might affect the user attentional state. This tracking may include the observation of the computing environment (e.g. the user being active in a different application) or it may report events generated by less common devices capable of recognising the phone ringing, a person entering the room, etc. Whilst it may be very difficult to use some of this information, knowledge of some specific environmental events may improve the ability of the agent to understand the behaviour of users switching between Atgentive-applications and "unknown" applications.

Idle application events report that the application has been idle longer than a *maximum idle application* time. The *maximum idle application* time is one of the parameters that the application associates to the *init* event. The handling of this event is similar to the handling of the *idle input event* (see section 4.3.1).

Copy and paste events and frequent shifts events - These events, which report respectively copy and paste or frequent shifts operations between Atgentive application's windows and windows in other applications are used in order to recognise windows in non-Atgentive applications that are part of the current task context. When these events are generated the agents use this information to make inferences on the user's activity (as exemplified in scenario 8) and may propose to update the task context of the current task to include the new window.

Scenario 8: Re-attracting distracted user's attention
Applied to: AtgentNet
The user is working at a high priority task on the platform: writing a posting that is due in a few hours. The tracking devices recognise that the user is frequently switching between the platform's window for the post-writing and the window of a document D in a word processor (not Atgentive enabled). The agents tentatively associate the word processing window to the context of the post-writing task. Another tracking device reports an *idle input event* on the post-writing focus. Although this event would normally give rise to an agents' intervention to re-attract the user's attention to the post-writing task, the agents recognise that the user is active in the word processor window for document D. Since this window is associated to the context of the post-writing task, the agents assume that the user is working at the task in another application window and do not intervene.

4.3.3 Community tracking

Community-tracking devices generate events reporting community changes that might affect the user attentional state. These events normally correspond to the recognition of patterns in the foci selected by members of the community or in metadata describing such foci. The information gained from community tracking events is used by agents to both inform individual users about the overall community activity and to make suggestions based on this activity.

Resource history events report on resources used by other users in the community just before and just after the resource just selected by the user; this events are inspired by recommender systems. For example, scenario 12 below is a simplified version of a recommendation method described by Chalmers (2000)

Scenario 12: Suggest community relevant resources
Applied to: AtgentNet
The system keeps track of the sequence in which all users open Knowledge Assests (KAs) in the platform. For every KA, a 'league table' is maintained of the KAs most frequently selected immediately both before and after the main KA (we will refer to each of these as a "related Knowledge Asset - rKA).
When a user selects a KA he/she will be offered the n (number to be determined) rKAs most likely to be of relevance in understanding the KA they chose (i.e. most temporally related).
To reduce the cost of interruption, the user will be offered the additional documents (rKAs) immediately upon selection of a KA. While the user may select one of the proffered rKAs (which will each open in an additional new window), no action need be taken by the user if he/she so choose.
When a KA is reopened, (i.e. after the first time for that user) the user will be offered the most frequently selected rKAs, as described above, AND any rKAs they accepted previous times for the current KA (if they do not now appear as the top n entries in the 'league table').
Once a KA has been selected n times without accepting the related KAs the agent will stop offering rKAs for that particular KA (but the user may ask for related KAs for that KA).

Community foci sequence events report the recognition of a path in the user activity that matches the activity of other users and allow agents to support task continuation as exemplified in scenario 13 below.

Scenario 13: Suggest community relevant tasks
Applied to: AtgentNet
The sequence of foci <"read D1 on the platform", "read D2 on the platform"> performed by the current users matches the beginning of the sequence <"read D1 on the platform", "read D2 on the platform", "reply to posting D3"> of 5 out of 6 other members of the community. The agents proposed to this user to continue his/her activity by performing "reply to posting D3".

4.3.4 Summary of Tracking Events

Table 4 gives examples of tracking-devices triggering events.

TRACKING EVENTS (Events generated by tracking devices)			
Event name	**Description**	**Example** **Comments**	**Possible parameters**
User tracking			
Idle-input event	User has not performed any input activity for longer than a given expected reaction time	No keyboard, nor mouse activity	
Low input frequency event	User is providing input at a rate slower than expected	Slow keyboard or mouse activity	
Foci sequences	The agent recognises a pattern in the sequence of foci	See scenario 11	
Low alertness event Low engagement event High stress event	Events generated by psycho-physiological measurements		
Environment tracking			
Idle application event	The application has been idle for a certain amount of time	The user has temporarily left the application	
Unknown application event	The user has started using an unrecognised application	A different application window becomes active If other applications are also interfaced with Atgentive the system may be able to recognise whether events from different applications relate to the same task or goal.	Note: no further discussed
Physical event event	Tracking devices report changes in the physical environment that may indicate a switch in the user's attentional state.	The phone rings, someone walks in the room	Note: no further discussed
Copy and past event	Reports copy and paste operations between the window(s) of the current task and other windows	Allows to associate windows from other applications to the context of the current task	Window
Frequent shifts event	Reports frequent shifts between the window(s) of the current task and other windows	Allows to associate windows from other applications to the context of the current task	Window
Community tracking			
Resource history event	Reports on resources used by other users in the community just before and just after the resource just selected by the user	May be used to create an historical task context	Resource list
Community foci sequence events	Report the recognition of a path in the user activity that matches the activity of other users	Support task continuation	Foci sequence

Table 4 – Events generated by the tracking devices

5 Conclusions

We have presented an event based conceptual framework for the support of attentional processes in learning and collaboration. This framework was based on two theoretical models. The first one describing how supporting learning at regulative, cognitive, and meta-cognitive levels results in redirecting student attention through specific interventions; the second one modelling such interventions at four further levels.

This analysis, which was based on realistic scenarios, has allowed us to define a taxonomy of events and interventions that would guide systems capable of supporting attentional processes. The application of the conceptual framework to the two pilots has demonstrated that such framework is both sufficiently general and complete, and therefore well suited as a basis for the design phase. The modularity of the proposed framework has several advantages:

1. It allows us to selectively implement parts of it within the project
2. It clearly defines a set of reference tasks (Whittaker, Terveen, & Nardi, 2000) against which different systems may be compared
3. It allows the larger research community to contribute further improvements

The formative evaluation has started by focussing on several scenarios and it will be detailed in the forthcoming deliverable D4.2 (Rudman & Zajicek, 2006). So far, the preliminary results seem to indicate that the scenarios 1, 9, 10, 11 are only partially relevant for AtgentSchool. Scenario 2 is possible when a planning module is added to Ontdeknet. Scenarios 3 and 4 (as well as 12, 13, 14) are very applicable. Scenarios 5 and 6 will need a user interaction module and further evaluation is needed to decide in what form they can be implemented. Finally scenarios 7, 8, and 15 are useful, but dependent on the possibilities of a listening module.

The results of this evaluation will be used to set the project's priorities in order to select which elements of the framework will be carried-on in the design and implementation phase whilst further evaluations will be made on simple prototypes.

6 Appendix 1 – Recent publication that gives an overview of the project

(Roda & Nabeth, 2006)

The AtGentive project: Attentive Agents for Collaborative Learners

Claudia Roda[1,3], Thierry Nabeth[2,3]

[1] American University of Paris, Computer Science Department,
112, rue du Bac, 75007 Paris, France
croda@aup.fr
[2] INSEAD Centre for Advanced Learning Technologies,
Bvd de Constance, 77305 Fontainebleau Cedex, France
thierry.nabeth@insead.edu
[3] AtGentive consortium
http://www.atgentive.com/

Abstract. Attention, which intervenes at many different levels such as the perception of the environment and the allocation of cognitive resources, appears to represent one of the key factors of learning and working performance. This poster presents AtGentive, a project which aims at investigating the use of agent-based ICT systems for supporting the management of the attention of young or adult learners in the context of learning and collaborative environments.

1 Introduction

Attention appears to represent one of the key factors of learning or working performance. The most effective learners and knowledge workers are often those people capable of filtering and selecting the most relevant information and at allocating their cognitive resources in the most appropriate manner.

This ability to manage efficiently attention can be considered as even more critical in the new learning and working contexts [see for example 1, 2, and 3]. For instance, in an online learning setting, the learners are more on their own; they have less guidance; and cannot situate themselves with others and adjust their behaviour. Even in presence of strong commitment, it is harder to evaluate optimal time allocation, and effectiveness of learning or collaborative processes. In this context, the learners have fewer points of reference to situate themselves, may not receive any direct pressure from a tutor or from their peers, and can more easily procrastinate, or engage into learning activities that are very ineffective. In the “knowledge economy”, employees are engaged in a multitude of projects involving a variety of actors from different horizons with which they have to collaborate. They also have to process more information and solicitation than in the past, originating from a variety of sources, and available in different forms (news, email, instant messaging, etc.). Finally, they are more autonomous and more responsible for their lines of actions. As a consequence, the knowledge workers have more risk to be overwhelmed by too much information and too many interruptions, and also to manage inefficiently the execution of the many tasks they have to accomplish for their work [1]. These new conditions typically results in a situation of information and cognitive overload for the learners and for the knowledge workers that represent a real challenge that need to be addressed so as to facilitate the setting up and the adoption of new methods of learning and work in the Information Society.

The AtGentive project is born from the idea that such a challenge can be addressed with the help of ICT systems that are aware of this attentional dimension, and that are able to support the individuals or the groups at filtering and selecting the most relevant information for them and helping them allocating their cognitive resources. Practically, the objective of this project is to investigate how to design such attention supporting systems, and in particular to explore how artificial agents, that may be embodied into artificial characters, can be used to provide more active and intelligent support to attention.

The first part of this poster will present the AtGentive project: what are the objectives of this project, what are the principles that will be used to design attention supporting systems for elearning and collaborative working context. The second part of this poster will present how these principles are going to be applied to design and to test "attention support" in two applications: (1) **AtGentSchool**: an elearning application for child education; (2) **AtGentNet**: an advanced virtual community platform supporting knowledge exchange of communities of distributed managers. The last part of this poster will conclude, and will indicate the future work.

2 A project aiming at supporting individual and group attention

2.1 An Overview of the AtGentive project

The AtGentive project explores the links between learning and attention and it includes: (1) the definition of a model of attention for the learner (low-level & high level, individual & social); (2) different mechanisms to capture or infer information about the state of attention (both at a low and high level) of the user in agent-enhanced collaborative learning platforms; (3) Intervention mechanisms using attentive agents informed about users attentional state, providing guidance, helping users to better manage their attention to achieve their learning objective, reducing information overload, and therefore improving the effectiveness of the learning & working process.

More specifically, the important foci of the project include:

- A model of attention of the user, and the support of this attention by attention informed systems. A deep understanding of attention (importance, impact, processes to support it, etc.) will be elaborated
- The "sensing" of the environment and in particular the different means for collecting information that can be used to profile the user's attentional state.

- The design of a set of mechanisms for supporting the management of the attention of users and groups (for instance enhancing the user perception).
- The design of an artificial agent cognitive architecture able to proactively and intelligently support the user attention.

Practically, the technical infrastructures that will be designed in this project will consist in ICT platforms that are enhanced with different components (intelligent or not) providing different level of attentional support such as:

- Components and approaches helping to enhance the perception of the user by filtering or emphasizing the information presented or delivered to the user, and therefore contributing to the reduction of information and cognitive overload. Examples of such approaches include the information design of the spaces facilitating the user perception, the use of visual tags emphasizing the most important items, or the ordering of the information according to the importance of the item (relevance for the user, freshness of the item, popularity of the item).
- Components and agents facilitating the organization of learning and work processes. For instance, this support can be provided for the execution of activities in presence of multitasking, frequent interruptions, and large information sources. Examples of such mechanism include enhanced notification mechanisms and interruption management (e.g. delaying a notification, or delivering it in a way that does not break the concentration of the user), and the support of task resumption (saving and restoring a context for a task that has been interrupted).
- Intelligent agents and mechanisms able to coach the learners or the knowledge worker in reaching higher level of performance in managing their attention. These agents will in particular help the user to acquire attention management skills. Examples of these agents and mechanisms can include the provision of statistics displaying the different activities of the user and the time he/she has allocated to them, but also proactive agent interventions making suggestions to the user (for instance the agent may suggest a new focus, or may encourage the user to adopt a more effective learning or communication practice)

2.2 A First Sketch of the AtGentive internal architecture

The agents, which may appear embedded in a artificial characters, are able to profile the learners' attentional state (short or long term) by observing their actions. On the basis of this profile, Atgentive agents can assess, analyze, and reason about user's attention, and provide assistance (for instance automating some tasks) and proactive coaching (assessment, guidance, stimulation, etc.).

Fig 1. represents a first sketch of the AtGentive architecture that will be further elaborated and then implemented during the project.

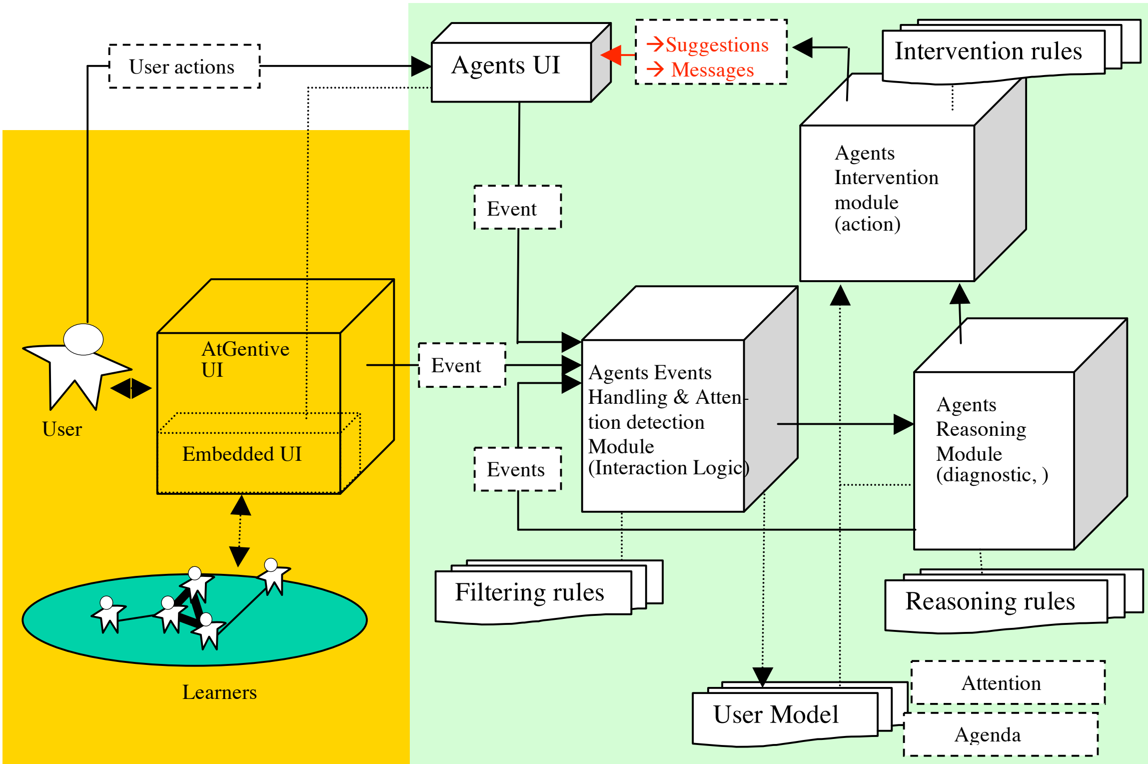


Fig. 1. A first sketch of the AtGentive internal architecture

The main components of this architecture are:

- The attention detection module which monitors and profiles the learner's attentional state.
- The attention reasoning module. This module represents the main intelligent reasoning part. Its function is to (1) Assess the situation; (2) Identify and build a selection of possible assistance and coaching interventions.
- The Intervention module. The function of this component is to select and execute the interventions that are the more appropriate (having the maximum impact on the user, and do not unnecessary distract the user).
- The interface. The user interface will rely on well designed interaction portals and artificial characters (artificial agents may intervene as embodied agents) that are attention "friendly".

3: AtGentSchool and AtGentNet: Two applications applying the AtGentive principles

Two different "attentive" applications are used to validate the principles elaborated in the AtGentive project: (1) **AtGentSchool**: an active elearning platform for child education; (2) **AtGentNet**: an advanced virtual community platform.

3.1 AtGentSchool: an "Attentive" active elearning platform for child education

AtGentSchool is an e-learning application for child education that will be built upon on the existing Ontdeknet system [4]. Ontdeknet is a learning environment founded on learning arrangement principles [5], an educational approach based on constructivism, situatedness, and active engagement into collaborative learning activities.

The incorporation of the support of attention is expected to improve the effectiveness of these systems by increasing the level of involvement of the learners, and leading to a more effective learning process. This objective will be accomplished by having the opportunity to measure and to assess the individual activity and attention patterns, during the learning arrangement, and later to allow adaptive adjustments to the learning objects and their interconnections.

A number of advantages are expected for the AtGentSchool users:

- The behaviour of the existing embodied agent (whose role is to help children through the learning process) will appear more natural to the user, which will enhance acceptance and prevent (the Microsoft dog) irritation.
- The agent behaviour will become adaptive, resulting in more efficient guidance
- Adaptation to the user is expected to enhance motivation
- More effective guidance and higher motivation are expected to lead to more effective learning experiences

3.2 AtGentNet: an "Attentive" advanced virtual community platform for supporting knowledge exchange

AtGentNet will consist in the design of an advanced virtual community platform supporting knowledge exchange of communities of distributed managers, relying on a model structuring of the communication using spaces currently implemented in the ICDT platform [6].

The incorporation of the support of attention in the ICDT platform will result in a set of mechanisms helping to enhance communication and knowledge exchange inside the platform.

Examples of mechanisms include:

- Advanced communication filtering and management mechanisms helping to reduce the knowledge and information overload associated with interaction with the other members of the community.
- Social awareness and translucence mechanisms displaying the activity happening in the community, helping to orient the attention of the users.
- Advanced monitoring capabilities (capturing the actions of the users, and basic filtering about these behaviours conducting to the generation of events that can be exploited by human or artificial agents)
- Artificial agents having some capacities to observe and to reason about the environment and about the users and to intervene proactively.
- "Attentional" expertise, allowing the agents to support managers in managing their own attentional state, both at the low level, and at the higher cognitive level.
- Artificial character interface, displaying agents as an anthropomorphic characters.

4 Conclusion and future work

The project has already started, with the involvement of the end users, to draft a series of scenarios. Such scenarios explore different ways to provide support for attention in an ICT platform. A few of the mechanisms have already been incorporated into first prototypes.

The AtGentive project is still at a too early stage to draw any definitive conclusions. However we have already observed that the concept of attention, by providing a new angle of analysis taking into account deeply human cognition, appears to be very useful to guide the design of mechanisms addressing the real issues that people are facing today (such as information overload, and the complexity of user-computer interaction). In some cases the concept of attention has provided us with a new angle of analysis of existing mechanisms that were not initially intended for supporting attention.

The second phase of this project, and its validation with the different pilots (one in schools, the second in a learning network), will help us to better understand which are the mechanisms supporting attention that are the most useful, and in particular will help us to assess the effectiveness of the new mechanisms (such as agent-based support) that will be elaborated for this projects.

Acknowledgments. The work described in this paper was partially sponsored by the EC under the FP6 framework project Atgentive IST-4-027529-STP. We would like to acknowledge the contribution of all project partners including: Laurent Ach, Albert A. Angehrn, Jaroslav Cech, Eugeni Gentchev, David Kingma, Pradeep Kumar Mittal, May Liem, Ivana Mala, Inge Molenaar, Koen Molenaar, Benoit Morel, Thierry Nabeth, Barbora Parrakova, Paul Rudman, Hari Siirtola, Toni Vanhala, Maurice Vereecken, Deng Ye, Mary Zajicek.

References

1. Davenport, Thomas and Beck John (2001); The Attention Economy; Harvard Business School Press, 2001
2. Roda, C., & Nabeth, T. (2005). The role of attention in the design of learning management systems. Proceedings IADIS International Conference CELDA (Cognition and Exploratory Learning in Digital Age), Lisbon, Portugal, 148 - 155.
3. Roda, C., & Thomas, J. (2006). Attention aware systems: Theories, applications, and research agenda. Computers in Human Behavior, 22(4), 557–587.
4. Ontdeknet: http://www.ontdeknet.nl/
5. Simons, P.R.J., Van der Linden, J.L., & Duffy, T. (Eds.) (2000). New learning. Dordrecht: Kluwer.
6. Albert A. Angehrn (2004); Designing Effective Virtual Communities Environments: The ICDT Platform; INSEAD CALT Report 10-2004

7 Appendix 2 – Study of Ondeknet's intervention model

Extracted from
The intervention model of Ontdeknet
Version: 0,8 draft
Authors: Inge Molenaar

Current situation Ontdeknet

Learning sequences

Lessons on Ontdeknet are always guided by the so-called learning sequences. The platform allows teachers and learning to the create sequences. A learning sequence is a number of activities a learner has perform on Ontdeknet to reach a certain learning goal. In a learning sequence two types of activities can be used: activities around assignments and activities around the experts.

Ontdeknet supports the use of the following assignment types:

- Written paper
- Visual paper
- Questionnaires with open and/or multiple choice questions

The activities that support the assignments are:

- Goal selection
- The mind map
- The expert

The activities that a student can perform in relation to the experts are:

- Introducing themselves
- Assigning to an expert
- Reading Dairies
- Posing questions
- Chatting with an experts
- Exploring the experts information

All the above activities can be selected a used in a learning sequence.

A example of a learning sequence The five step

This learning sequence consists of five steps, which are an analogy of the normal process of collaboration. The five process steps are:

Introduction
The student introduces himself through filling out his personal information.
The expert introduces himself through filling out his personal information and writing an introductory story.
Goal setting
The student sets his goals for working with Ontdeknet.
The experts takes notice of the goal
Assigning to an expert
The student assigns him/herself to the expert who can support him to reach the goal.
The expert receives a notification that a new student has arrived
Making the mind map; mutual understanding
The student describes the topics he would like to address in the collaboration with the expert by entering these topics in a mind map. The student can use the introduction dairy and the

personal information of the expert to obtain the necessary context for filling out the mind map. The expert receives the mind map to see what the student likes to learn from him.

The assignment

The student reads the contributions of the expert and asks questions.

The expert will write a contribution on each of the topics, typically one per week. He will answer questions, which provides the opportunity to further collaborate on certain issues.

The tools in the five step

Different tools in the environment support this process:

Step	Tools		
User	Student	Expert	Teacher
Introduction	Personal information sheet	Personal information sheet	Monitor
Goal setting	Goal set sheet	Project information	Provide assignment
Assigning to an experts	Search expert by reading their personal information and introduction diary	Write introduction dairy	Monitor
Setting topics	Mind map	Overview Mind map items	Monitor
Work on paper	Read contributions Ask questions Write paper	Write contributions Answer question	Monitor

Table 2. Tools in the Ontdeknet environment to support the five step process.

Onty, the Agent

Currently, the reactions of the agent are connected to the advancements of the students in respect to the activated learning sequence and the student's position of the platform. Templates are created for this purpose. Around very activity on Ontdeknet a templates is built. A template contains the following elements:

1. An introduction, what is this activity?
2. Navigation guidance's, where do you perform this activity?
3. Explanation, how do you perform this activity? (activated upon arrival at right screen)
4. Finalising, this activity is finished

In the templates navigation guidance is included when an user moves in the wrong direction he is redirected by the agent. Based on the mouse actions of the student different parts of the template are activated. The system registers the situation on the platform and the advancements of the students. The systems decides on the basis on advancements of the user, if a template is started and it registers where the sequence will start depend on the situation on the platform of the user.

Adaptive adjustments to these templates are possible, but demand for interpreting activities of the teacher and are very work intense. The AtGentive approach would allow for a more efficient process of adaptive interaction with the agent, without necessary involvement of the teacher.

The inventarisation

In the following research setting the problems with the current approach are made explicit through an analysis of student activities working on the five step. Per step the current interventions are described. Problems that students are experiencing are analyzed in the cases and possible intervention solutions are provided.

The research setting

The five step has been used to support the assignment to write a paper about a profession with the help of an expert. This has been tested at a school for primary education. Each group has been be assigned to the learning sequence the five step. The groups have been working in a computer lab for 1 hour per week under supervision of the researcher. The groups worked for 6 weeks on their tasks. Three students were working together on one computer. The children received an instruction for the activities for that lesson at the beginning of the hour and are then asked to work as independent as possible. The researcher observed the groups to asses if they understood the task and to observe, if interesting occurrences take place. Notes were made during the observation. During their work the groups were recorded with a microphone to obtain protocols of their conversation.

Sample

We have used 4 groups of 3 students. 2 groups of students were formed with students from group 5(8 years old) and 2 groups of students consisted of students from group 8(12 years old). The groups were formed by the teachers based on the following guidelines to ensure group members with diverse backgrounds:
The three students who have three different school advises for high school education

- There is always one student with a good reading capabilities
- There is always one student with good computer skills
- There is a mix of boys and girls in the groups

Task setting

This pilot will provide insights in the activities in the learning sequence and the reactions to the agent. The focus of this research will be on the reactions to the agent and the extra interventions needed.

Measurements

Process measurements: protocol analysis
We obtained protocols of students of two different age groups working on the learning sequence the five steps. In the analysis of the protocols the following occurrences were taken out:

- are the interventions causing the anticipated reaction of the children
- were in the protocol do we need additional interventions of the teacher

Results

Below an analysis of step 1 until steps 4 is provided.

Description of step 1: the introduction

Activity	Intervention sort	Intervention	Intervention type
Introduce yourself	Introduction	I am Onty. I would like to know who you are. Will you introduce ourself to me?	MC: orientation
	Navigation 1	Click on my workingcorner	Pd: Navigation 1
	Navigation 2	Click on about me	Pd: Navigation 2
	Explanation	You are on the right spot to introduce yourself. Shall I show you how to introduce yourself? I am Onty… Now it is your turn	C: explanation
	Finish activity	Nice to get to know you!	MC: evaluation
	Feedback unfinished	You are not done yet, please continue with introducing yourself.	MC: execution

The cases

Studiegroepje (8 years old):
Students follow Onty, they understand the explanation immediately and start to introduce themselves.

1a. After a while they do not know what to add anymore
Intervention teacher [content]: the teacher suggests writing their hobbies and age.
Case Atgentive: [cognitive intervention]:Atgentive monitors low activity, Onty comes up with

1b. Finish: "What else can we write? Nothing we are done"
Case Atgentive: notices finish activity → asks if they are done→ provides feedback

DJT(8 years old):
The students follow Onty, they understand the explanation and start by dividing tasks (You think, I write and you pose the questions). They start with introducing themselves.

2a. They forget to enter a title
Intervention teacher [procedure]: "you should include a title, for instance the name of your group"
Case Atgentive: notices no title→ suggests a title

2b. Finish: The students ask the teacher: Are we done?
Intervention teacher [procedure]: "Yes do not forget to save!"
Case Atgentive: Are we done? → a checklist is shown, did you discuss all topics? →
Yes? Do not forget to save.

Bloopers (11 years old)
3a. Students are making jokes while reading what Onty says. Then they do not know what they have to do.
Students; What do we have to do?
Intervention teacher [procedure]: what did Onty just say?
Students: that he is 666 years old..
Intervention teacher [procedure]: what did he want to know?
Students: Click on him again, aha introduce yourself to your expert.
Now students start to work
Case Atgentive: notices that students are lost → He ask do you know what to do? → Student responds with no → Atgentive starts to explain

3b. Finish: Intervention teacher [procedure]: time is up, please save.

Ontix (11 years old)
4a. Students did not start the learning line but just started to surf on Ontdeknet.
Intervention teacher[process]: “what are you doing?”
Student:: I do not know
Intervention teacher [procedure]: start the learning line XX
Case Atgentive: notices that students are surfing without starting a learning line → Asks: What are you doing? → Redirects students to starting a learning line.

Finish: Students finish by saving

Activity	Input	Intervention sort	Intervention type	Intervention
Case 1a	task: introduction Students: new users Activities: inactivity Time on task: 15 minute Amount of text in fields: large	Cognitive	C support	“Did you include the following topics?” Shows a checklist
Case 1b	task: introduction Students: new users Activities: finish Time on task: 15 minute Amount of text in fields: large	Metacognitive	MC monitoring	“Quite a story your wrote, good to meet you!.

Case 2a	Task: introduction Students: new users Activities: entering information Time on task: 5 minutes Amount of text in fields: empty title field	Regulative	R missing information	"You could fill in the tittle,, now"
Case 2b.	task: introduction Students: new users Activities: activate agent Time on task: 15 minute Amount of text in fields: large	Students ask: are we done? Metacognitive	MC monitoring	Shows checklist, did you included all these topics?
Case 3a.	task: introduction Students: new users Activities: activate agent Time on task: 2 minutes Amount of text in fields: empty	Students ask: what should we do? Metacognitive	MC orientation	Introduction of the activity
Case 3b.	task: introduction Students: new users Activities: typing text Time on task: 30 minutes, times is up	Regulative	R	Please save time is up.

	Amount of text in fields: medium			
Case 4a	task: introduction Students: new users Activities: wrong screens Time on task: 0 minutes Amount of text in fields: empty	Regulative	R	Please activate a learning sequences to start learning

Intervention schema activity introduction

Introduction

Problem	Intervention group	Intervention sort	Intervention type	Description	Parameters	Text
Navigational problem, first step O1	Regulation	Navigatie1	RN1	Navigation support on the first layer	new activity for user type of user (level 1-3) inactivity x time when not on activity screen wrong direction	Click on my working corner I will first click on my working corner here you introduce yourself
Navigational problem, second step O2	Regulation	Navigatie 2	RN2	Navigation support on the second layer	new activity for user inactivity x time when not on activity screen wrong direction	Click on about me
Navigational problem, third step O3	Regulation	Navigatie 3	RN3	Navigation support on the third layer	new activity for user inactivity x time when not on activity screen wrong direction	Click on change
Retract focus O4	Regulation	Iac	RIac	Redirect to the task	inactivity for x time when on activity screen	Hello are you still here?
Missing information O5	Regulation	MI	RMI	Missing information on task	- empty box	I miss information on XX
Times up O6	Regulation	Tu	Rtu	Session time is up	- time length has past	You Ontdeknet session is ending, please save you work!
Activate learning sequence O7	Regulation	LS	Rls	Activate learning sequence	- no learning sequences activated	Please activate learning sequence
Start task O8	Cognition	Explanation	Ce	Explain the task	- new activity for user - inactivity for x time when on	In the introduction you tell more about yourself. I will give an

					activity screen	example……
Continue/ resmue task O9	Cognition	Continuation	CC	Continue	- loging in	You were bussy with introducing yourself will you now continue?
Supply information O10	Cognition	Support	Cs	Supply direct information to continue the task	information in activity field: N, S, M, L. new or low profile user	No information: you can discus the following topics in your introductio Small information: did you think to discuss the following topics Medium information: did you discuss all topics? Large information: almost done, dic you include all topics?
Start subtask/ proposed event within a task O11	Cognition	Direction	Cd	Supply information to locate additional information	information in activity field advanced user	Check the section about me of our experts to compare you introductior with the one of the experts
New information/ focuss O12	Cogntition	New information	Cnf	There is new relevant information for the user	- information relevant for the task	- your expert just introduced himsel
Learning sequence O13	Metacognition	Orientation	MCo	Introduction of the learning sequence and the goal	long new user short old user	Long: On ontdeknet you work with learning sequence. <…> Short: This is your learning sequence <view>
Proposed task/ Proposed	Metacognition	Planning	MCp	Planned time	planning parallel/ serieel time in session	You have X time for this activity. This activity needs to be finished by XX

focus event beyond a task O14						You need to sequence these activities
Support event O15	Metacognition	Execution	MCe	Information on activity	- activate when there is no cognitive support possible	-
Complete event O16	Metacognition	Monitoring	MCm	Feedback on the activity	-finished activity - activate based on amount of input information in activity field: N, S, M, L.	N: you have not filled in any information please try again S: It seems a very short introductior are you sure you are done? M: Nice to meet you L: Quite a story, I will sit down to read this
Complete event O17	Metacognition	Evaluation	MCev	Feedback Activities planning relationship	- feedback when concluding the session, q, r, s	Q: you did this activity very fast R: you were right on track S: you ook you to long to introduce yourself
Complete event O18	Metacognition	Reflection	MCr	Feedback on learning	- after completion new activity	Now you learned to introduce yourself

User suggestions not yet included

Description of step 2: setting a learning goal

Activity	Intervention sort	Intervention	Intervention type
Introduce yourself	Introduction	I would like to know what you want to learn on Ontdeknet?.....	MC: orientation
	Navigation 1	Click on my workingcorner	Pd: Navigation 1
	Navigation 2	Click on assignment	Pd: Navigation 2
	Navigation 3	Click on the right assignment	Pd: Navigation 3
	Navigation 4	Click on subject	Pd Navigation
	Explanation	You are on the right spot to fill in your learning goal. For instance I want to learn all about onty… ha ha joke. Good luck…	C: explanation
	Finish activity	Now I know what you want to learn	MC: evaluation
	Feedback unfinished	You are not done yet, please continue to write you learning goal.	MC: execution

The cases

Studiegroepje (8 years old):
2a. Students start to work on their learning goal. Onty says click on the <tabheader>
Intervention teacher [content]: what is a tabblad? T: shows tabblad
Case Atgentive: Students ask “what is a tabblad in question box, answers….”

2b. Students are having a discussion about the topic of their paper.
Intervention teacher [content]: your assignment is about the fishfarmer, not about fish.
Student: “see that is what I said”.
Case atgentive; X

2c. Lianne is play with Onty while beng is trying to type text
Intervention teacher [procedure]: Lianne when you play with the mouse Beng can not type at the same time.
Case Agtentive: please do one activity at the same time!!

DJT(8 years old):
No problems

Bloopers (11 years old)
2d. Students write their goal are satisfied quickly and continue on the next step; “This is it, save”

Intervention teacher [process]: are you done already?
Case Agtentive: notices little time spend on the task. Provides suggestions and check on the work.

Activity	Input	Intervention sort	Intervention type	Intervention
Case 2a	task: set learning goal Students: new users Activities: inactivity Time on task: 0 minute Amount of text in fields: none	Cognitive	C support	"a tabheader is….
Case 2b	task: set learning goal Students: new users Activities: inactivity Time on task: 0 minute Amount of text in fields: none	Cognitive	C explain	Impossible to intervene
Case 2c	Task: introduction Students: new users Activities: entering information and playing with onty Time on task: 5 minutes Amount of text in fields: little text	Regulative	R impossible action	"You can not do 2 things at the same time"
Case 2d	Task: introduction Students:	Metacognitive	MC monitoring	Shows checklist, did you included all these topics?

	new users Activities: entering information Time on task: 2 minutes Amount of text in fields: little text			

Intervention schema activity introduction: setting a learning goal

Problem	Intervention group	Intervention sort	Intervention type	Description	Parameters	Text
Navigational problem, first step O19	Regulation	Navigatie1	RN1	Navigation support on the first layer	new activity for user type of user (level 1-3) inactivity x time when not on activity screen wrong direction	Click on my working corner… I will first click on my working corner… here you set your learning goal
Navigational problem, second step O20	Regulation	Navigatie 2	RN2	Navigation support on the second layer	new activity for user inactivity x time when not on activity screen wrong direction	Click on assignments
Navigational problem, third step O21	Regulation	Navigatie 3	RN3	Navigation support on the third layer	new activity for user inactivity x time when not on activity screen wrong direction	Click on the right assignment
Navigational problem, n step O22	Regulation	Navigatie n	RN…n	Navigation support on the third layer	new activity for user inactivity x time when not on activity screen wrong direction	….
Retract focus O23	Regulation	Iac	RIac	Redirect to the task	inactivity for x time when on activity screen	Hello are you still here?
Missing information O24	Regulation	MI	RMI	Missing information on task	- empty box	I miss information on XX
Times up O25	Regulation	Tu	Rtu	Session time is up	- time length has past	You Ontdeknet session is ending, please save you work!
Regulative impossible action	Regulation	Rim	Rim	The user is performing an action	-	You can not do this

O26				that is not possible		
Activate learning sequence O27	Regulation	LS	Rls	Activate learning sequence	- no learning sequences activated	Please activate learning sequence
Start task O28	Cognition	Explanation	Ce	Explain the task	- new activity for user - inactivity for x time when on activity screen	You set a learning goal to explain tc the expert what you want to learn
Continue/ resume task O29	Cognition	Continuation	CC	Continue	- logging in	You were busy with writing your learning goal will you now continue?
Supply information O30	Cognition	Support	Cs	Supply direct information to continue the task	information in activity field: N, S, M, L. new or low profile user	No information: you can discus the what you want to learn in the learning goal Small information: did you think to discuss the following topics Medium information: did you discuss all topics? Large information: almost done, dic you include all topics?
Start subtask/ proposed event within a task O31	Cognition	Direction	Cd	Supply information to locate additional information	information in activity field advanced user	Check the section “what I will discuss” in the information of the expert
New information/ focuses O32	Cognition	New information	Cnf	There is new relevant information	- information relevant for the task	x

				for the user		
Explain difficult word O33	Cognition	Explanation word	Cexw	A difficult word is explained		This means….
Learning sequence O34	Metacognition	Orientation	MCo	Introduction of the learning sequence and the goal	long new user short old user	Long: On ontdeknet you work with learning sequence. <…> Short: This is your learning sequence <view>
Proposed task/ Proposed focus event beyond a task O35	Metacognition	Planning	MCp	Planned time	planning parallel/ serieel time in session	You have X time for this activity. This activity needs to be finished by XX You need to sequence these activities
Support event O36	Metacognition	Execution	MCe	Information on activity	- activate when there is no cognitive support possible	-
Complete event O37	Metacognition	Monitoring	MCm	Feedback on the activity	-finished activity - activate based on amount of input information in activity field: N, S, M, L.	N: you have not filled in any information please try again S: It seems a very short learning goal, are you sure you are done? M: good to know what you want to learn L: Quite a story, I will sit down to read this
Complete event O38	Metacognition	Evaluation	MCev	Feedback Activities planning	- feedback when concluding the session, q, r, s	Q: you did this activity very fast R: you were right on track S: you ook you to long to introduce

				relationship		yourself
Complete event O39	Metacognition	Reflection	MCr	Feedback on learning	- after completion new activity	Now you learned to set a learning goal

User suggestions not yet included

Description of step 3: find your expert

Activity	Intervention sort	Intervention	Intervention type
Find your expert	Introduction	On Ontdeknet you can work with experts…	MC: orientation
	Navigation 1	Click on my expert	Pd: Navigation 1
	Explanation	You are on the right spot to find your expert. Fill in a search term and click on search	C: explanation
	Support	Do you see the expert you are looking for, click on the experts	C: support
	Support	Is this the right expert for you? Click on support	C: support
	Finish activity	Now you have an experts	MC: evaluation
	Feedback unfinished	You are not done yet, please find your expert	MC: execution

The cases

Studiegroepje (8 years old):
3a. Students follow onty to search module, but have trouble finding the right search term (search engine is very restrictive) to find the fishfarmer.
Teacher intervention [content]: fish and water is the name of the expert
Case Atgentive: Do you need help to find the right search term?

3b. Students have trouble to continue. They do not understand what they need to do.
Teacher intervention [navigation]: You have to click on the picture
Case Atgentive: notices trouble students to continue, “You have to click on the picture”

DJT(8 years old):
3c. Students follow onty and experience a problem with going to the experts. The expert is below the search graph.
Teacher intervention [navigation] You have to scroll down and select an expert.
Case Atgentive; have to scroll down and select an expert.

Activity	Input	Intervention sort	Intervention type	Intervention
Case 3a	task: find an expert Students: new users Activities: fill in many	Cognitive	C support	“do you need hel to find the right search term?

	search terms Time on task: 10 minute Amount of text in fields: many terms			
Case 3b	task: find an expert Students: new users Activities: fillëd in a search term Time on task: 2 minute Amount of text in fields: a list of results	regulative	R navigation	You have to click on the picture
Case 3c	task: find an expert Students: new users Activities: fillëd in a search term Time on task: 2 minute Amount of text in fields: a list of results	Regulative	R navigation	“You need to scroll down to see your results”

Intervention schema activity introduction: Finding an expert

Problem	Intervention group	Intervention sort	Intervention type	Description	Parameters	Text
Navigational problem, first step	Regulation	Navigatie1	RN1	Navigation support on the first layer	new activity for user type of user (level 1-3) inactivity x time when not on activity screen wrong direction	Click on my working corner… I will first click on my working corner… here you search your expert
Navigational problem, second step	Regulation	Navigatie 2	RN2	Navigation support on the second layer	new activity for user inactivity x time when not on activity screen wrong direction	Click my expert
Navigational problem, n step	Regulation	Navigatie n	RN…n	Navigation support on the third layer	new activity for user inactivity x time when not on activity screen wrong direction	….
Retract focus	Regulation	Iac	RIac	Redirect to the task	inactivity for x time when on activity screen	Hello are you still here?
Missing information	Regulation	MI	RMI	Missing information on task	- empty box	I miss information on XX
Times up	Regulation	Tu	Rtu	Session time is up	- time length has past	You Ontdeknet session is ending, please save you work!
Regulative impossible action	Regulation	Rim	Rim	The user is performing an action that is not possible	-	You can not do this
Activate learning	Regulation	LS	Rls	Activate learning	- no learning sequences activated	Please activate learning sequence

sequence				sequence		
Start task	Cognition	Explanation	Ce	Explain the task	- new activity for user - inactivity for x time when on activity screen	You find an experts by filling a search term…
Continue/ resume task	Cognition	Continuation	CC	Continue	- logging in	You were busy with searching an expert will you now continue?
Supply information	Cognition	Support	Cs	Supply direct information to continue the task	Hits on the search term: N, S, M, L. new or low profile user	No information: You need to fill in search term Few hits: please try another search term, Medium hits: Do you find your expert here? Many hits: you have a lot to choice form?
Start subtask/ proposed event within a task	Cognition	Direction	Cd	Supply information to locate additional information	information in activity field advanced user	Check the section "what I will discuss" in the information of the expert to decided if this expert is useful
New information/ focuses	Cognition	New information	Cnf	There is new relevant information for the user	- information relevant for the task	X
Explain difficult word	Cognition	Explanation word	Cexw	A difficult word is explained		This means….
Learning sequence	Metacognition	Orientation	MCo	Introduction of the learning	long new user short old user	Long: On ontdeknet you work with learning sequence. <…> Short: This is your learning

				sequence and the goal		sequence <view>
Proposed task/ Proposed focus event beyond a task	Metacognition	Planning	MCp	Planned time	planning parallel/ serieel time in session	You have X time for this activity. This activity needs to be finished by XX You need to sequence these activities
Support event	Metacognition	Execution	MCe	Information on activity	- activate when there is no cognitive support possible	-
Complete event	Metacognition	Monitoring	MCm	Feedback on the activity	-finished activity -	You have found an expert
Complete event	Metacognition	Evaluation	MCev	Feedback Activities planning relationship	- feedback when concluding the session, q, r, s	Q: you did this activity very fast R: you were right on track S: you ook you to long to introduce yourself
Complete event	Metacognition	Reflection	MCr	Feedback on learning	- after completion new activity	Now you found an expert

User suggestions not yet included

Description of step 4: making the mind map

Activity	Intervention sort	Intervention	Intervention type
making the mind map	Introduction	Step 4 is very important, we are going to make a mind map	MC: orientation
	Navigation 1	Click on my workplace	Pd: Navigation 1
	Navigation 2	Click on assignments	Pd: navigation 2
	Navigation 3	Choice the right assignment	Pd Navigation 3
	Navigation 4	Click on mind map	Pd: Navigation 4
	Navigation 5	Click on change	Pd: navigation 5
	Explanation	The mind map is my magic cure to find out what you already know……. Give it a try, what do you know about this topic	C: explanation
	Support	In every box you type a related word, fill the empty boxes	C: support
	support	In the info the topic of the expert is explained, here you can read everything about your expert….	C: support tip
	Finish activity	Good job, your mind map is read! Now we are starting with the real work, shall I explain step 5?	MC: evaluation
	Feedback unfinished	You are not done yet, please fill in our mind map	MC: execution

The cases

Studiegroep (8 years old)
Students find the mind map quickly and start to fill it in.

4a. They seem to have problem with filling the second box.
Teacher intervention, [procedural]: “Just go to the next box, click on it”
Case Atgentive: [regulative intervention]: Notice problems switching between the boxes→ click the next box

4b. There is a second step in the mind mapping activity: first students brainstorm about all related topic. Second student select the topics they like to learn more about.
Teacher intervention, [content]: explanation of the second step in mind mapping: the selection of words you want to learn more about
Case Atgentive, [cognitive support] monitors the number of boxes filled and the time to fill them→ provide tips on proceeding → explains the next step if enough boxes are filled

4c. Finish: Students discuss if they know more association words for the mindmap, they then decide that they are done.
4c.1 Case Atgentive: [meta cognitive; monitoring] → provides feedback on outward issues times spend number of boxes filled
4c.2 Case Atgentive: [cognitive] → provides feedback related to the information entered in the mindmap

Activity	Input	Intervention sort	Intervention type	Intervention
Case 4a	Task: mind map Students; new users Activities; numerous clicks Time on task: 3 minutes Amount of text in box; 1 amount of boxes filled in: 1	Regulative	Regulative	Click on the next box
Case 4b	Task: mind map Students; new users Activities; inactivity Time on task: 13 minutes Amount of text in box; 1-2 amount of boxes filled in: 10	Meta cognitive	Orientation 2	Please continue with the selection of words you want to learn more about
Case 4c.1	Task: mind map	cognitive	feedback	The topics you selected are

	Students; new users Activities; inactivity Time on task: 20 minutes Amount of text in box; 1-2 amount of boxes filled in: 10 Boxes checked: 3			discussed by our expert Students who said X also said Y… Ect.
Case 4c	Task: mind map Students; new users Activities; inactivity Time on task: 20 minutes Amount of text in box; 1-2 amount of boxes filled in: 10 Boxes checked: 3	Meta cognitive	feedback	You have selected only X boxes

Intervention schema activity introduction: making a mind map

Problem	Intervention group	Intervention sort	Intervention type	Description	Parameters	Text
Navigational problem, first step	Regulation	Navigatie1	RN1	Navigation support on the first layer	new activity for user type of user (level 1-3) inactivity x time when not on activity screen wrong direction	Click on my working corner… I will first click on my working corner… here you fill in your mind map
Navigational problem, second step	Regulation	Navigatie 2	RN2	Navigation support on the second layer	new activity for user inactivity x time when not on activity screen wrong direction	Click assignments
Navigational problem, n step	Regulation	Navigatie n	RN…n	Navigation support on the third layer	new activity for user inactivity x time when not on activity screen wrong direction	….
Retract focus	Regulation	Iac	RIac	Redirect to the task	inactivity for x time when on activity screen	Hello are you still here?
Missing information	Regulation	MI	RMI	Missing information on task	- empty box	You did not fill in any informatior
Times up	Regulation	Tu	Rtu	Session time is up	- time length has past	You Ontdeknet session is ending, please save you work!
Regulative impossible action	Regulation	Rim	Rim	The user is performing an action that is not possible	-	You can not do this
Help	Regulation	Regulation help	RH	The user needs help	new user inactive time	Click on the next box

				on the interface	much clickin	
Activate learning sequence	Regulation	LS	Rls	Activate learning sequence	- no learning sequences activated	Please activate learning sequence
Start task	Cognition	Explanation	Ce	Explain the task	- new activity for user - inactivity for x time when on activity screen	The mind map is my magic cure t find out what you already know……. Give it a try, what do you know about this topic
Continue/ resume task	Cognition	Continuation	CC	Continue	- logging in	You were busy with filling in a mind map will you now continue?
Supply information	Cognition	Support	Cs	Supply direct information to continue the task	Hits on the search term: N, S, M, L. new or low profile user	No information: You need to fill i topics you want to learn more about Few hits: please fill in more topics Medium hits: Do you know more? Many hits: shall we continue with the second phase
Start subtask/ proposed event within a task	Cognition	Direction	Cd	Supply information to locate additional information	information in activity field advanced user	You can read the information of the expert…
New information/ focuses	Cognition	New information	Cnf	There is new relevant information for the user	- information relevant for the task	x
Feedback on our topics	Cognition	Feedback	CFB	Feedback on the content the	- when this is possible	The topics you selected are discussed by our expert

				student filled in		
Explain difficult word	Cognition	Explanation word	Cexw	A difficult word is explained		This means….
Learning sequence	Metacognition	Orientation	MCo	Introduction of the learning sequence and the goal	long new user short old user	Long: On ontdeknet you work wit a learning sequence. <…> Short: This is your learning sequence <view>
Continue in second phase	Metacogntivition	Orientation 2	MCO2	Introduction of the second phase in the step	atleast X boxes filled in inactivity	Please continue with the selection of words you want to learn more about
Proposed task/ Proposed focus event beyond a task	Metacognition	Planning	MCp	Planned time	planning parallel/ serieel time in session	You have X time for this activity. This activity needs to be finished by XX You need to sequence these activities
Support event	Metacognition	Execution	MCe	Information on activity	- activate when there is no cognitive support possible	-
Complete event	Metacognition	Monitoring	MCm	Feedback on the activity	-finished activity -	You have selected X boxes
Complete event	Metacognition	Evaluation	MCev	Feedback Activities planning relationship	- feedback when concluding the session, q, r, s	Q: you did this activity very fast R: you were right on track S: you ook you to long to introduce yourself
Complete	Metacognition	Reflection	MCr	Feedback	- after completion new activity	Now your expert knows what you

event				on learning		want to learn
Complete event	Metacognition	Reflection continueation	MCrc	Feedback on result of this activity	- after completion new activity	There will be a new dairy soon!

User suggestions not yet included

Description of step 5: the assignment
Step 5 is really a compilation of steps that the students need to perform. Currently Onty does not support this step, due to the complications with the linearity of Onty.

The student is supposed to perform the following activities on a weekly basis:

- read a dairy
- summarize the dairy
- write chapter of the paper
- formulate questions for the expert
- read answer of the expert
- added answers in the chapter

The process is followed for every chapter of the paper. The process does not need to be followed linearly. There are templates for read dairy, write chapter, pose question.

Allocation of activities

Atgentive needs to find out which order of activities is logical to follow based upon the information available in the student and expert environment:
What topics does the student want to learn more about?
On which topics are dairy available, of more which one is the best to start with?
Are there questions answers since the student was last online?
Has the student been working on a chapter?
Based on the analyses Atgentive provides the “right” next activity to perform or provides the student with the choice of activity to perform (depends on the level of the student).

Interventions

Once the activity is decided upon the interventions related to these activities can be activated. Intervention schema’s that need to be worked out are:

- read dairy
- summarize dairy
- write chapter
- formulate question
- read answer
- added answers in chapter

The cases
Students have not worked with Onty in this section of the research.

General intervention schema.
Sort of the intervention message is central to this intervention schema.

- Regulative
- Cognitive
- Metacognitive

Based on the above described intervention schema's per step we can derive a general intervention schema. <<to be done>>

The purpose of the wizard of Oz is to validate the completeness and the exclusiveness of this intervention schema as described in the introduction.

Additional intervention modalities

Intervention forms additionally to the above described intervention schema differ in features of the agent and the user control over the intervention

Features of the agent:

- appearance of the agent
- voice of the agent
- emotions of the agent
- size of the agent

User control over interventions:

- preferences
- demand for interventions
- acceptance of interventions
- feedback on interventions

These list are a first inventarisation and are not exclusive. <<To be done>>

8 Appendix 3 – Visual demonstration of how the Atgentive system may handle scenarios 1 and 9

This document is a PDF print out of a Power Point presentation available on the Atgentive platform.

Atgentive

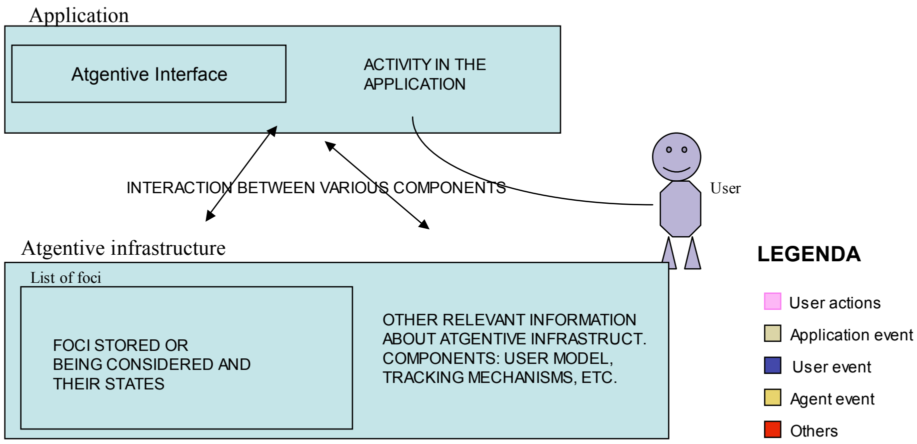
Demonstration of the
Event driven conceptual
framework

Claudia Roda - American University of Paris - 20.3.2006

ALL SCENARIOS ARE REPRESENTED BY A SET OF SLIDES ORGANISED AS FOLLOWS:

TITLE OF SCENARIO

TEXT DESCRIBING PARTS OF THE SCENARIO REPRESENTED IN PREVIOUS SLIDES, PART OF SCENARIO BEING REPRESENTED BY THIS SLIDE. THE REST OF THE SCENARIO



SCENARIOS

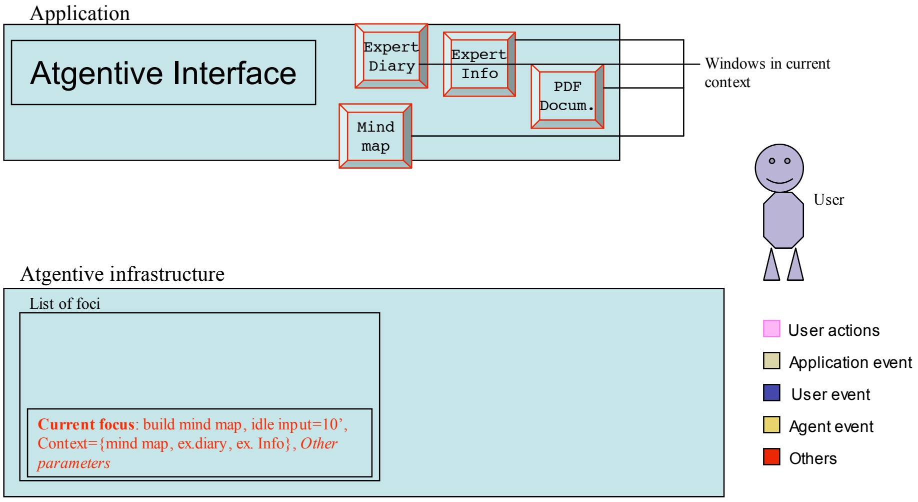
- S1: Support to task resumption
- S2: Support to limited time resources allocation

- Sample interface level interaction involving scenarios 1 and 2

Scenario 1
Support to task resumption, restoring task context

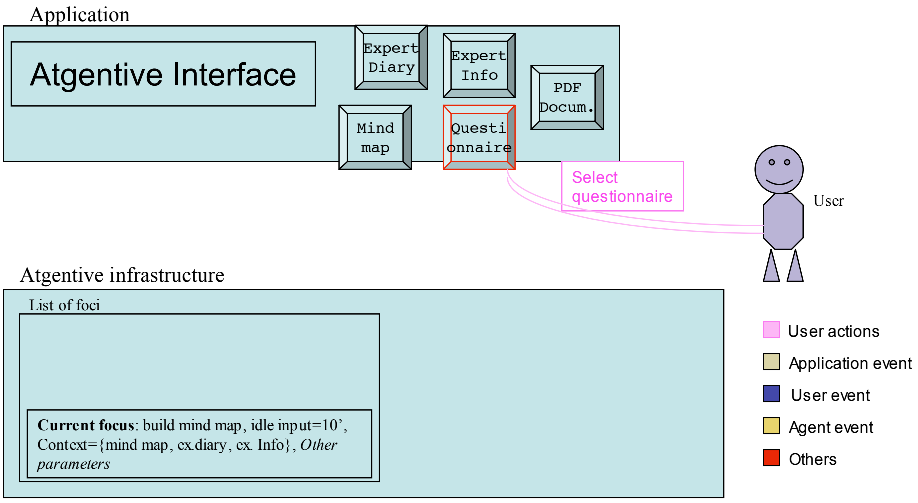
Scenario 1 Support to task resumption, restoring task context

The student is building the mind map (current focus) using the expert's introduction diary and personal information, as well as a pdf document opened in an Acrobat window. The student switches to a questionnaire. The Ontdeknet application reports a start event for the new activity (questionnaire). The agent saves, possibly with the help of the user, the context of the previous focus (which includes the mind-map window, the diary and information of the expert, and the pdf document window). Later the student returns to the mind-map building activity; the application sends a resume event; the agent proposes to restore the saved context.



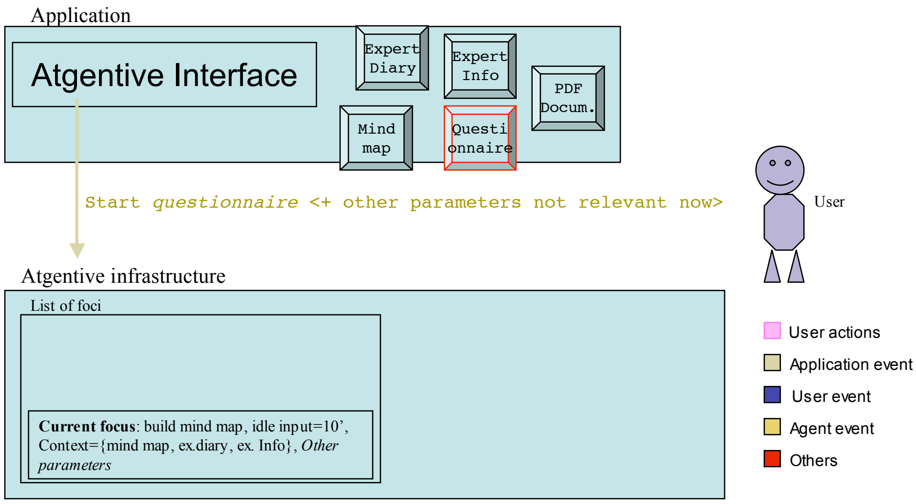
Scenario 1 Support to task resumption, restoring task context

The student is building the mind map (current focus) using the expert's introduction diary and personal information, as well as a pdf document opened in an Acrobat window. The student switches to a questionnaire. The Ontdeknet application reports a start event for the new activity (questionnaire). The agent saves, possibly with the help of the user, the context of the previous focus (which includes the mind-map window, the diary and information of the expert, and the pdf document window). Later the student returns to the mind-map building activity; the application sends a resume event; the agent proposes to restore the saved context.



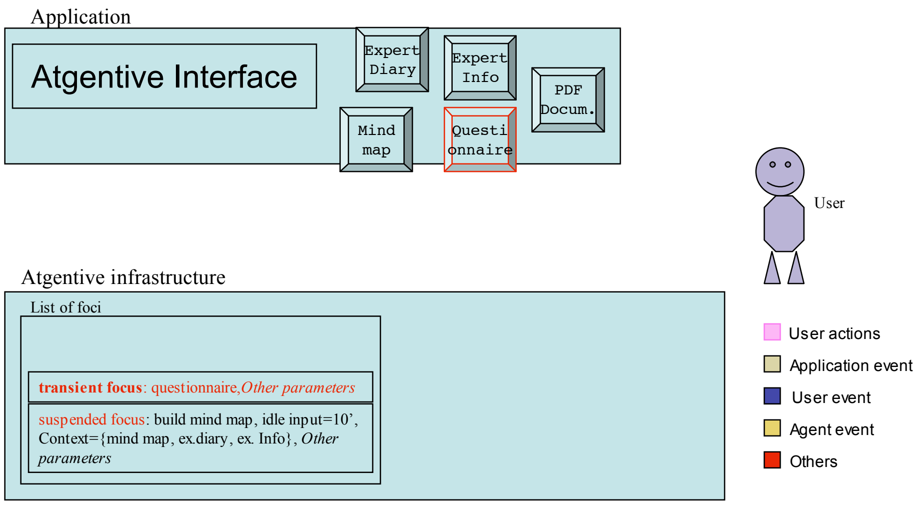
Scenario 1 Support to task resumption, restoring task context

The student is building the mind map (current focus) using the expert's introduction diary and personal information, as well as a pdf document opened in an Acrobat window. The student switches to a questionnaire. The Ontdeknet application reports a start event for the new activity (questionnaire). The agent saves, possibly with the help of the user, the context of the previous focus (which includes the mind-map window, the diary and information of the expert, and the pdf document window). Later the student returns to the mind-map building activity; the application sends a resume event; the agent proposes to restore the saved context.



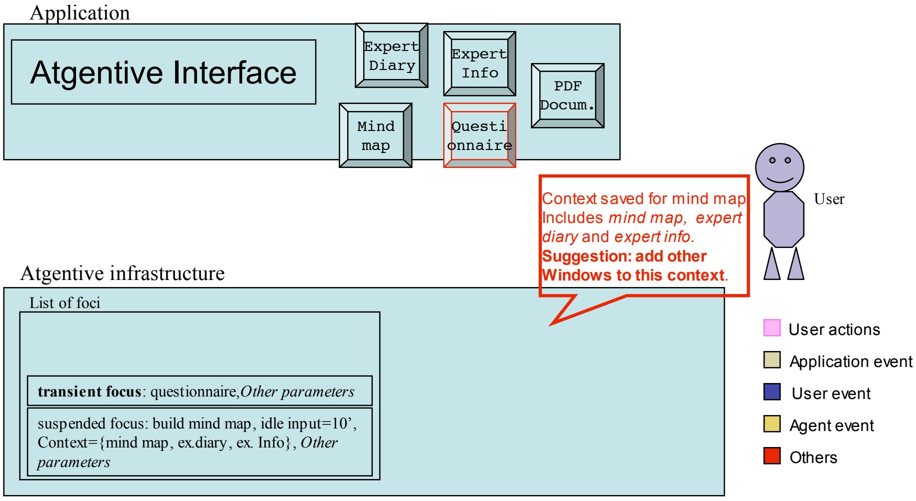
Scenario 1 Support to task resumption, restoring task context

The student is building the mind map (current focus) using the expert's introduction diary and personal information, as well as a pdf document opened in an Acrobat window. The student switches to a questionnaire. The Ontdeknet application reports a start event for the new activity (questionnaire). The agent saves, possibly with the help of the user, the context of the previous focus (which includes the mind-map window, the diary and information of the expert, and the pdf document window). Later the student returns to the mind-map building activity; the application sends a resume event; the agent proposes to restore the saved context.



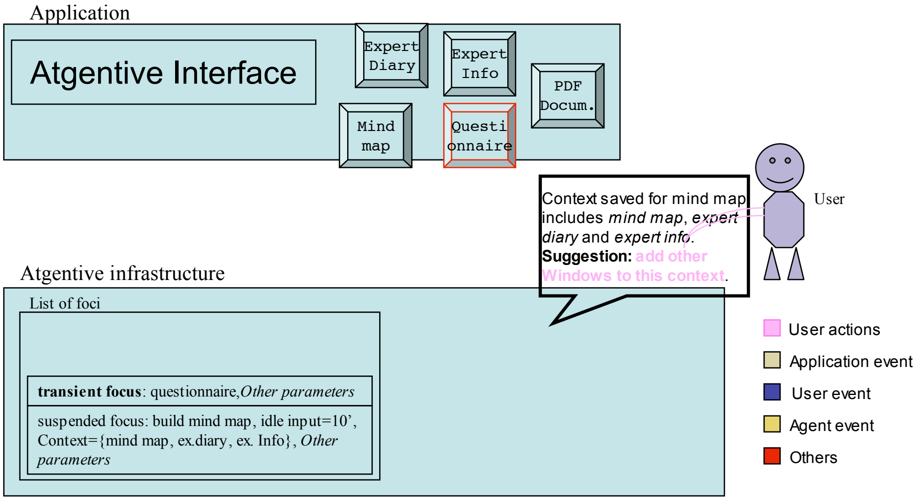
Scenario 1 Support to task resumption, restoring task context

The student is building the mind map (current focus) using the expert's introduction diary and personal information, as well as a pdf document opened in an Acrobat window. The student switches to a questionnaire. The Ontdeknet application reports a start event for the new activity (questionnaire). The agent saves, possibly with the help of the user, the context of the previous focus (which includes the mind-map window, the diary and information of the expert, and the pdf document window). Later the student returns to the mind-map building activity; the application sends a resume event; the agent proposes to restore the saved context.



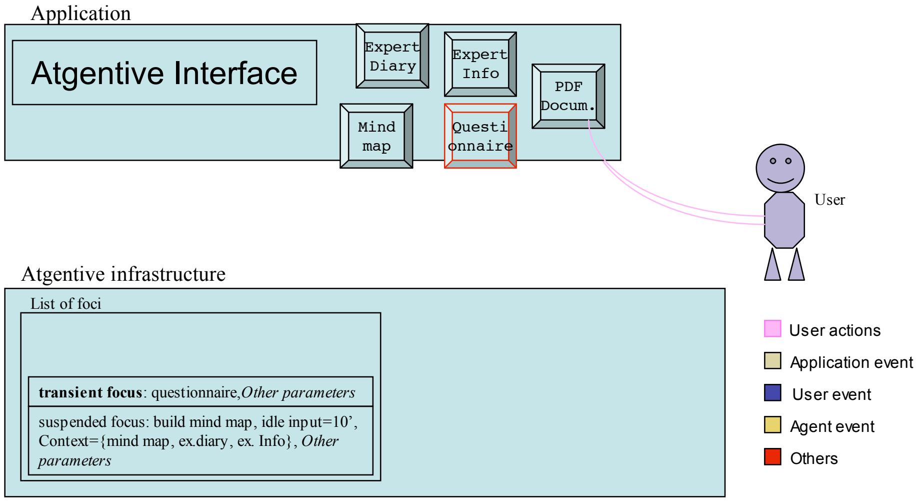
Scenario 1 Support to task resumption, restoring task context

The student is building the mind map (current focus) using the expert's introduction diary and personal information, as well as a pdf document opened in an Acrobat window. The student switches to a questionnaire. The Ontdeknet application reports a start event for the new activity (questionnaire). The agent saves, possibly with the help of the user, the context of the previous focus (which includes the mind-map window, the diary and information of the expert, and the pdf document window). Later the student returns to the mind-map building activity; the application sends a resume event; the agent proposes to restore the saved context.



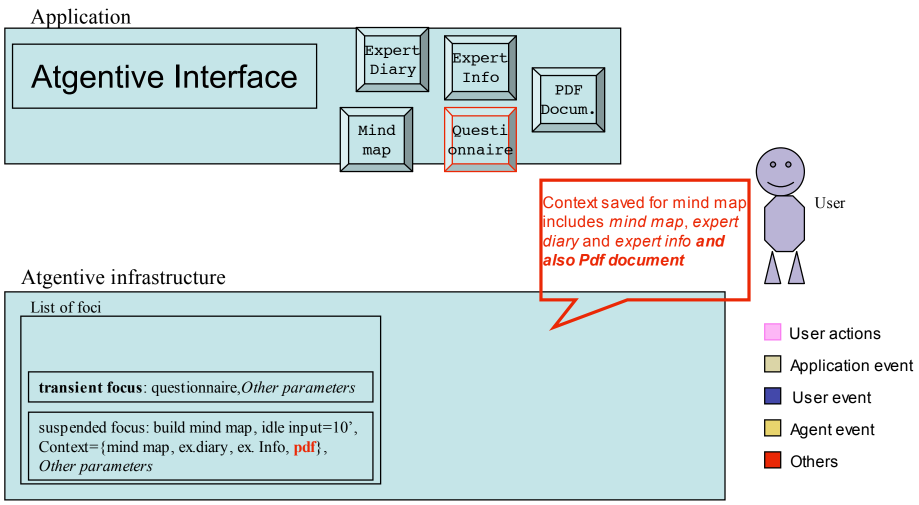
Scenario 1 Support to task resumption, restoring task context

The student is building the mind map (current focus) using the expert's introduction diary and personal information, as well as a pdf document opened in an Acrobat window. The student switches to a questionnaire. The Ontdeknet application reports a start event for the new activity (questionnaire). The agent saves, possibly with the help of the user, the context of the previous focus (which includes the mind-map window, the diary and information of the expert, and the pdf document window). Later the student returns to the mind-map building activity; the application sends a resume event; the agent proposes to restore the saved context.



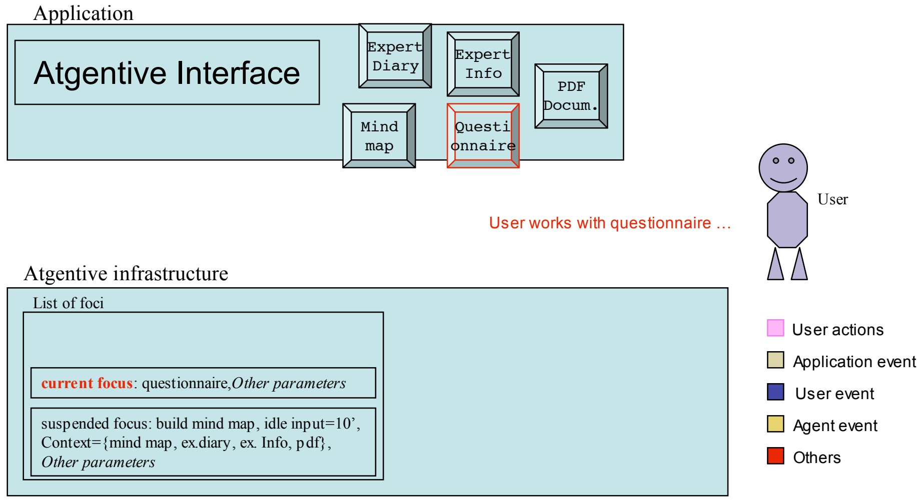
Scenario 1 Support to task resumption, restoring task context

The student is building the mind map (current focus) using the expert's introduction diary and personal information, as well as a pdf document opened in an Acrobat window. The student switches to a questionnaire. The Ontdeknet application reports a start event for the new activity (questionnaire). The agent saves, possibly with the help of the user, the context of the previous focus (which includes the mind-map window, the diary and information of the expert, and the pdf document window). Later the student returns to the mind-map building activity; the application sends a resume event; the agent proposes to restore the saved context.



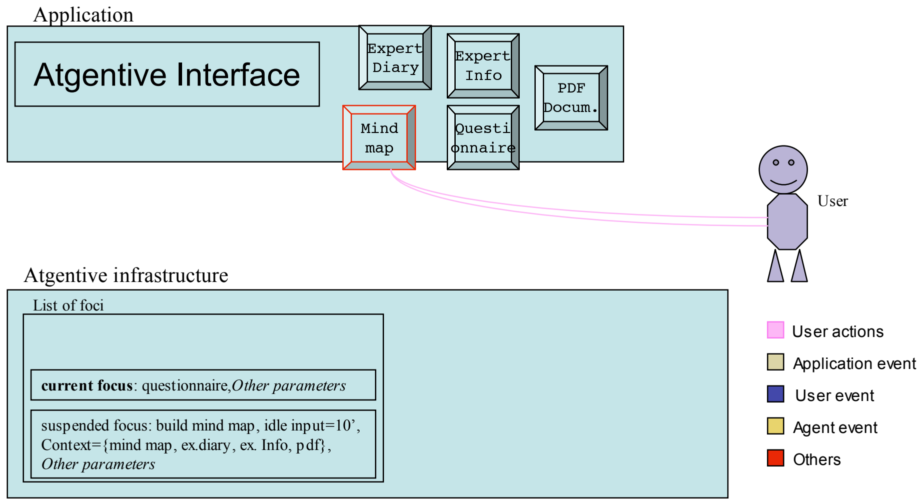
Scenario 1 Support to task resumption, restoring task context

The student is building the mind map (current focus) using the expert's introduction diary and personal information, as well as a pdf document opened in an Acrobat window. The student switches to a questionnaire. The Ontdeknet application reports a start event for the new activity (questionnaire). The agent saves, possibly with the help of the user, the context of the previous focus (which includes the mind-map window, the diary and information of the expert, and the pdf document window). Later the student returns to the mind-map building activity; the application sends a resume event; the agent proposes to restore the saved context.



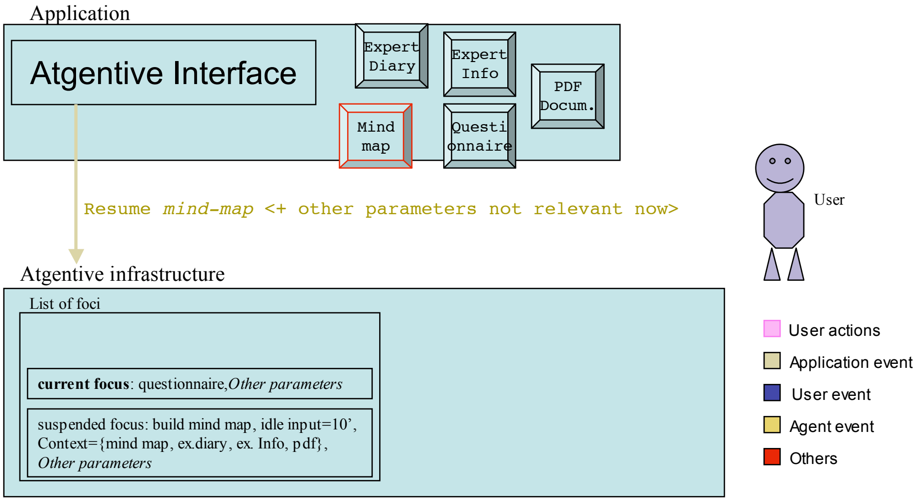
Scenario 1 Support to task resumption, restoring task context

The student is building the mind map (current focus) using the expert's introduction diary and personal information, as well as a pdf document opened in an Acrobat window. The student switches to a questionnaire. The Ontdeknet application reports a start event for the new activity (questionnaire). The agent saves, possibly with the help of the user, the context of the previous focus (which includes the mind-map window, the diary and information of the expert, and the pdf document window). Later the student returns to the mind-map building activity; the application sends a resume event; the agent proposes to restore the saved context.



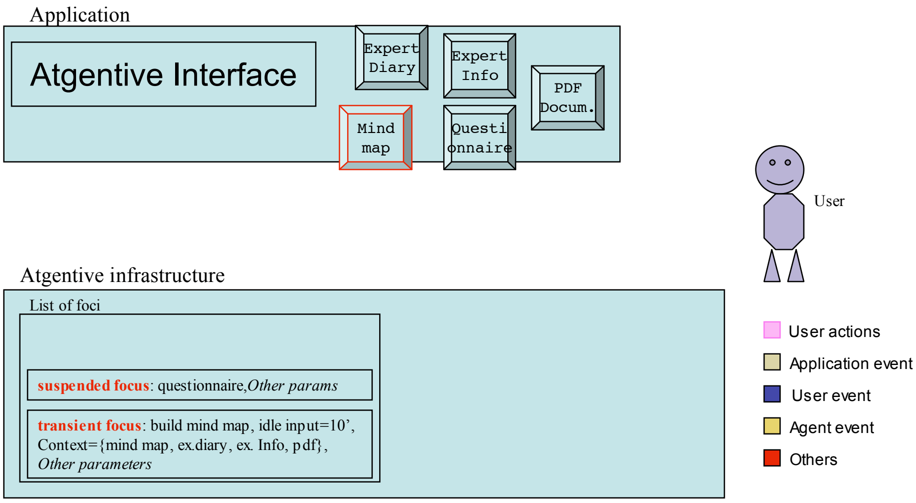
Scenario 1 Support to task resumption, restoring task context

The student is building the mind map (current focus) using the expert's introduction diary and personal information, as well as a pdf document opened in an Acrobat window. The student switches to a questionnaire. The Ontdeknet application reports a start event for the new activity (questionnaire). The agent saves, possibly with the help of the user, the context of the previous focus (which includes the mind-map window, the diary and information of the expert, and the pdf document window). Later the student returns to the mind-map building activity; the application sends a resume event; the agent proposes to restore the saved context.



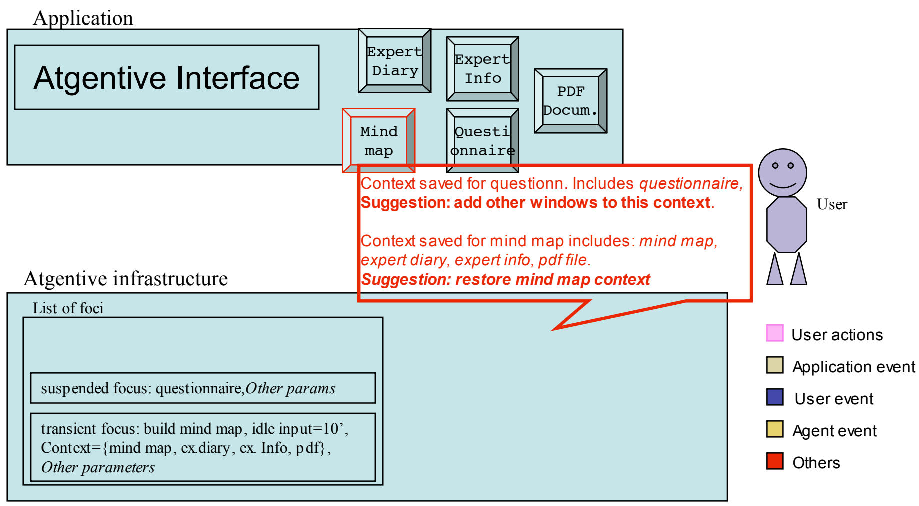
Scenario 1 Support to task resumption, restoring task context

The student is building the mind map (current focus) using the expert's introduction diary and personal information, as well as a pdf document opened in an Acrobat window. The student switches to a questionnaire. The Ontdeknet application reports a start event for the new activity (questionnaire). The agent saves, possibly with the help of the user, the context of the previous focus (which includes the mind-map window, the diary and information of the expert, and the pdf document window). Later the student returns to the mind-map building activity; the application sends a resume event; the agent proposes to restore the saved context.

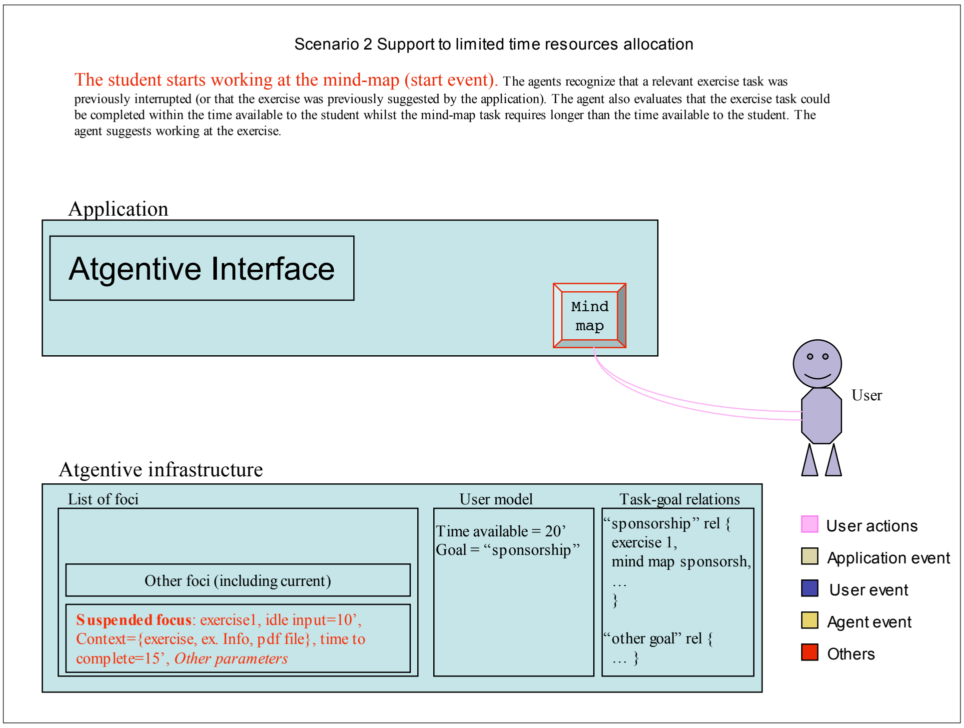


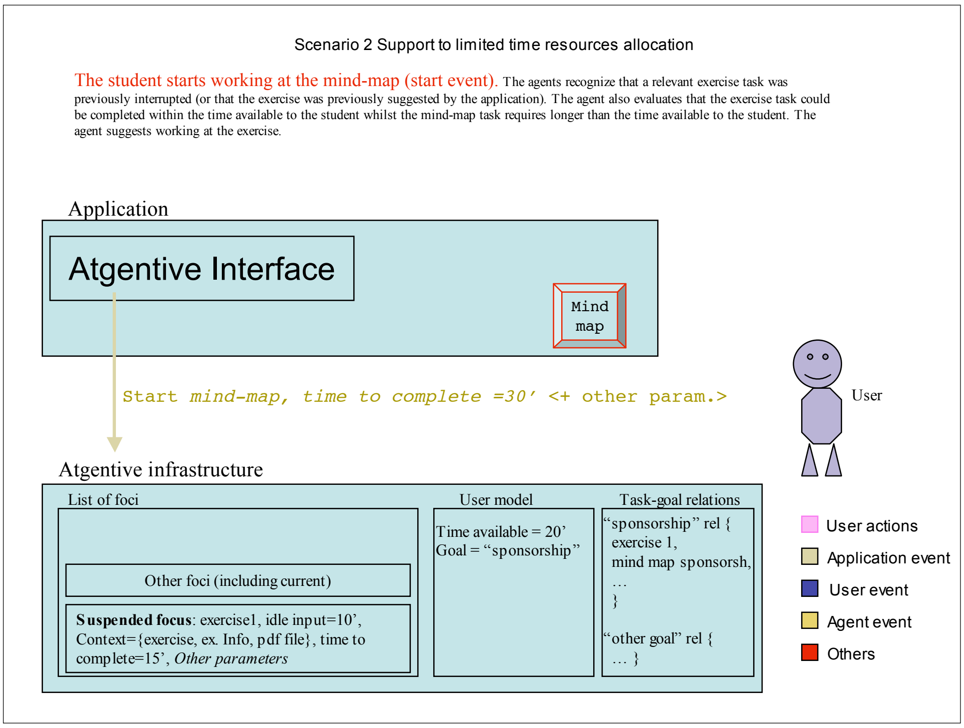
Scenario 1 Support to task resumption, restoring task context

The student is building the mind map (current focus) using the expert's introduction diary and personal information, as well as a pdf document opened in an Acrobat window. The student switches to a questionnaire. The Ontdeknet application reports a start event for the new activity (questionnaire). The agent saves, possibly with the help of the user, the context of the previous focus (which includes the mind-map window, the diary and information of the expert, and the pdf document window). Later the student returns to the mind-map building activity; the application sends a resume event; the agent proposes to restore the saved context.



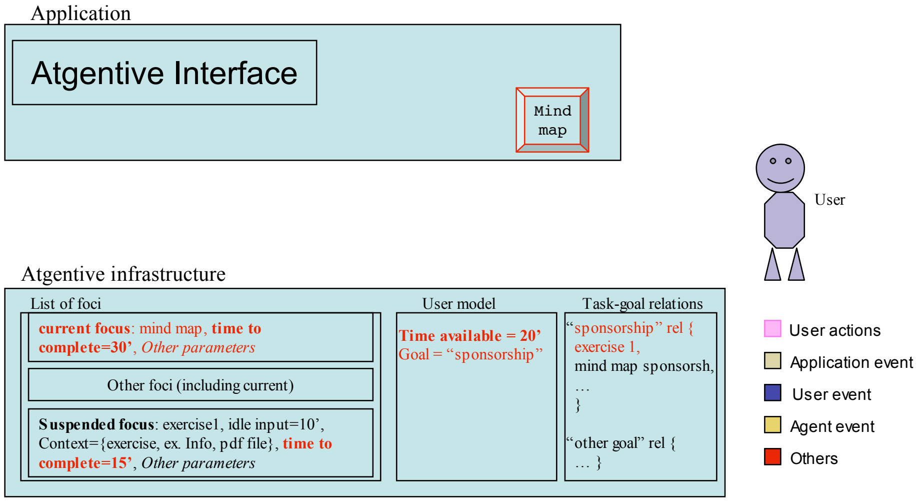
Scenario 2
Support to limited time resources allocation


Scenario 2 Support to limited time resources allocation
The student starts working at the mind-map (start event). The agents recognize that a relevant exercise task was previously interrupted (or that the exercise was previously suggested by the application). The agent also evaluates that the exercise task could be completed within the time available to the student whilst the mind-map task requires longer than the time available to the student. The agent suggests working at the exercise.
Application
Atgentive Interface
Mind map
User
Atgentive infrastructure
List of foci
Other foci (including current)
Suspended focus: exercise1, idle input=10', Context={exercise, ex. Info, pdf file}, time to complete=15', Other parameters
User model
Time available = 20'
Goal = "sponsorship"
Task-goal relations
"sponsorship" rel {
exercise 1,
mind map sponsorsh,
…
}
"other goal" rel {
… }
User actions
Application event
User event
Agent event
Others


Scenario 2 Support to limited time resources allocation
The student starts working at the mind-map (start event). The agents recognize that a relevant exercise task was previously interrupted (or that the exercise was previously suggested by the application). The agent also evaluates that the exercise task could be completed within the time available to the student whilst the mind-map task requires longer than the time available to the student. The agent suggests working at the exercise.
Application
Atgentive Interface
Mind map
Start mind-map, time to complete =30' <+ other param.>
User
Atgentive infrastructure
List of foci
Other foci (including current)
Suspended focus: exercise1, idle input=10', Context={exercise, ex. Info, pdf file}, time to complete=15', Other parameters
User model
Time available = 20'
Goal = "sponsorship"
Task-goal relations
"sponsorship" rel {
exercise 1,
mind map sponsorsh,
…
}
"other goal" rel {
… }
User actions
Application event
User event
Agent event
Others

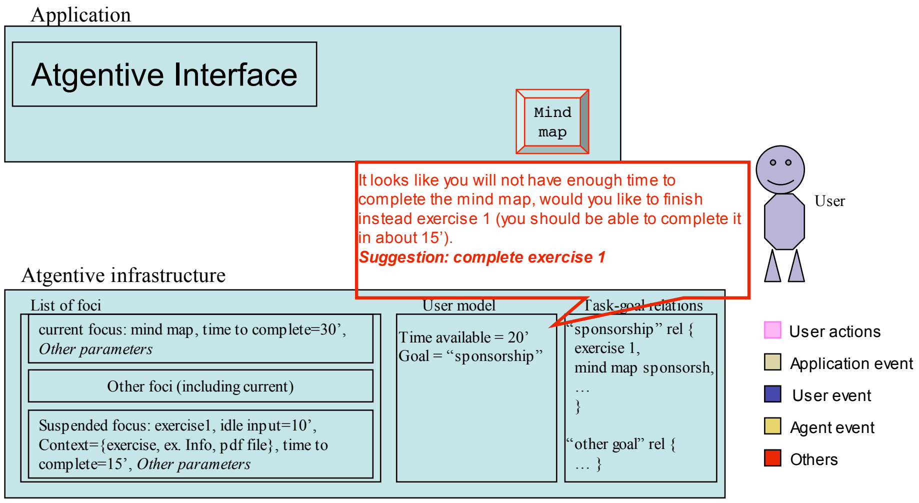
Scenario 2 Support to limited time resources allocation

The student starts working at the mind-map (start event). The agents recognize that a relevant exercise task was previously interrupted (or that the exercise was previously suggested by the application). The agent also evaluates that the exercise task could be completed within the time available to the student whilst the mind-map task requires longer than the time available to the student. The agent suggests working at the exercise.



Scenario 2 Support to limited time resources allocation

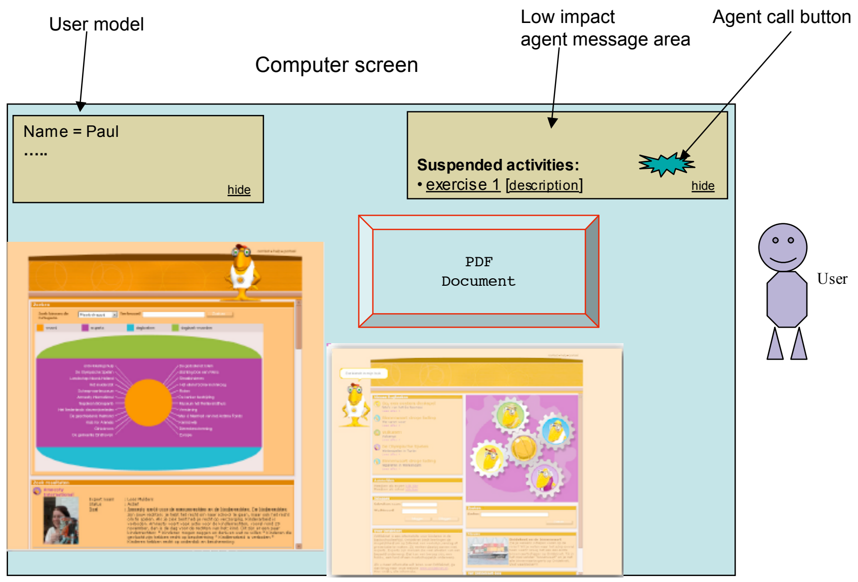
The student starts working at the mind-map (start event). The agents recognize that a relevant exercise task was previously interrupted (or that the exercise was previously suggested by the application). The agent also evaluates that the exercise task could be completed within the time available to the student whilst the mind-map task requires longer than the time available to the student. The agent suggests working at the exercise.



Sample interface level interaction involving scenarios 1 and 2

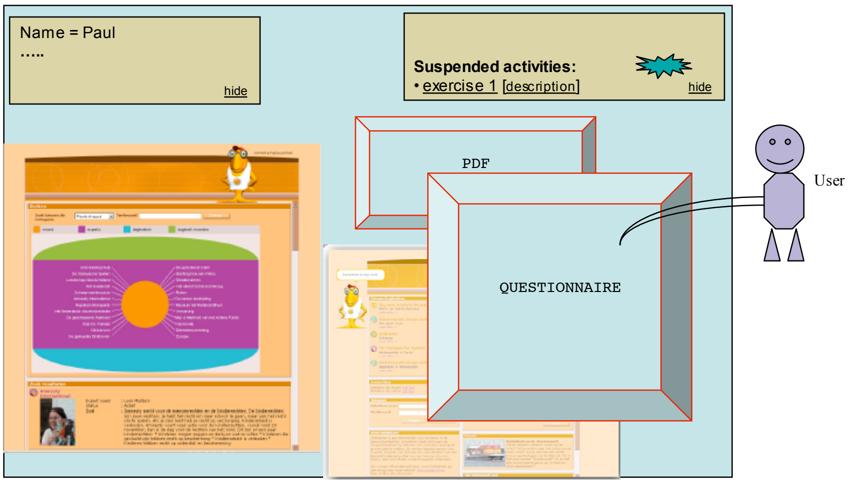
Possible interface level interaction involving scenarios 1 and 2

The student is working at the mind map.



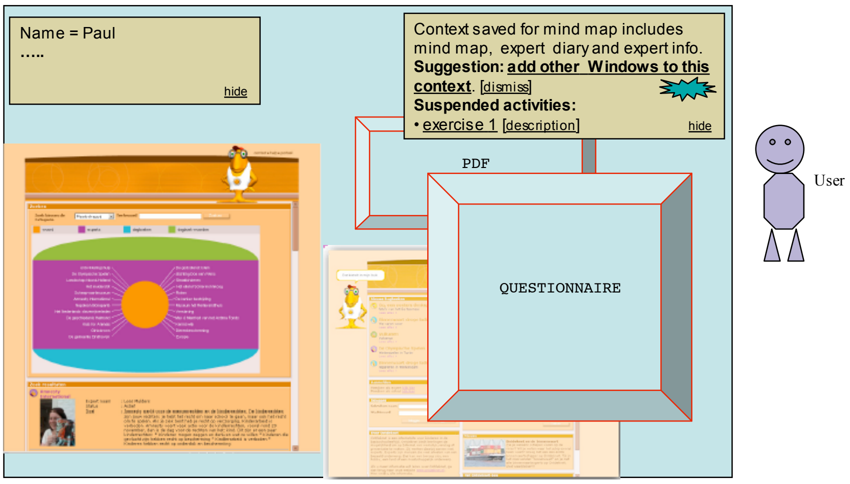
Possible interface level interaction involving scenarios 1 and 2

Student selects questionnaire



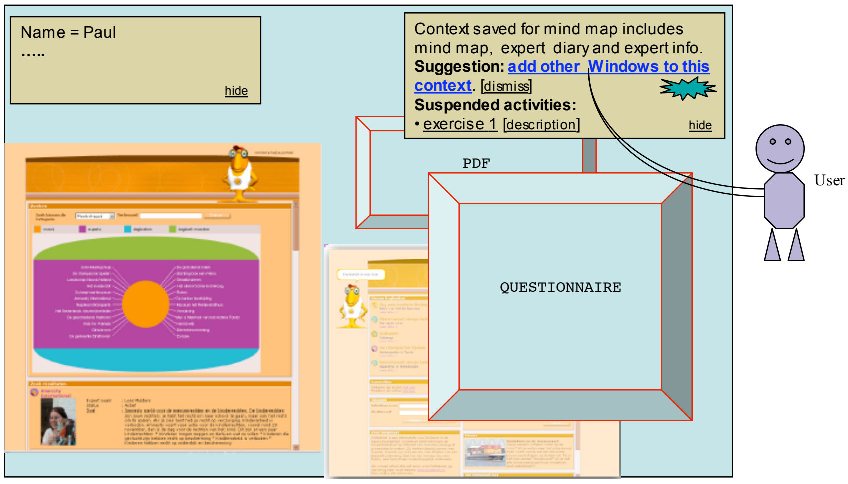
Possible interface level interaction involving scenarios 1 and 2

Agents requests support to save mind map context



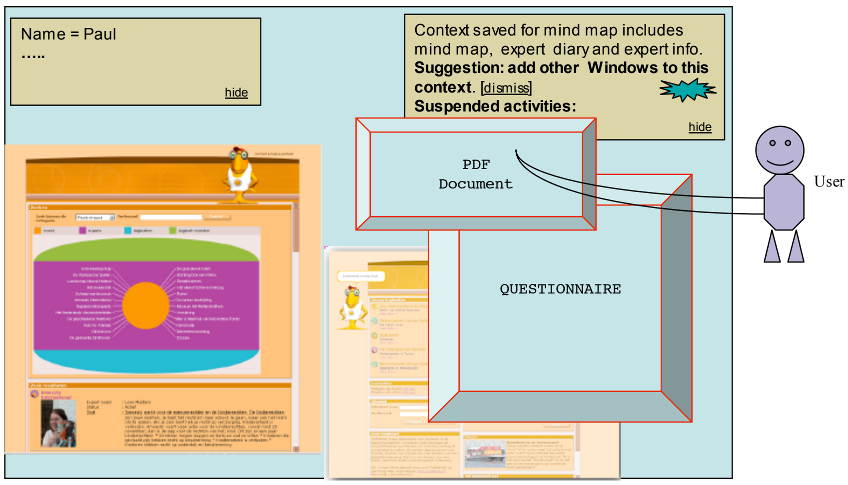
Possible interface level interaction involving scenarios 1 and 2

User accepts suggestion …



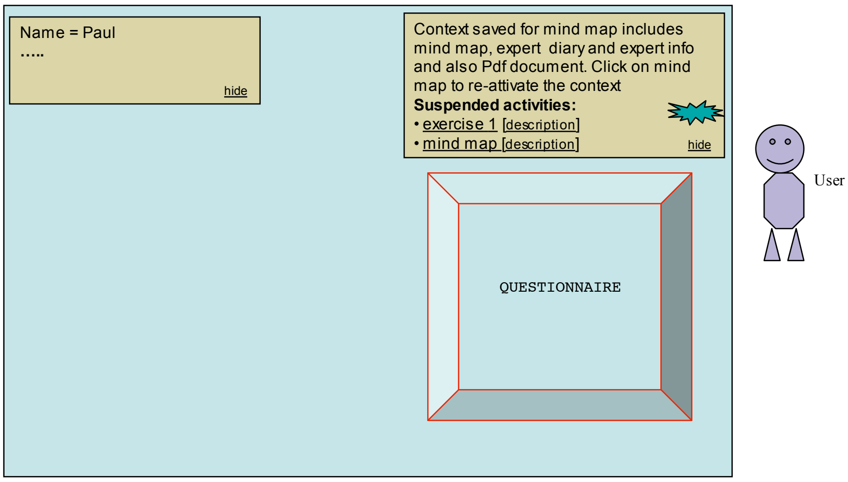
Possible interface level interaction involving scenarios 1 and 2

and selects the PDF files window as a window to add to the mind-map context



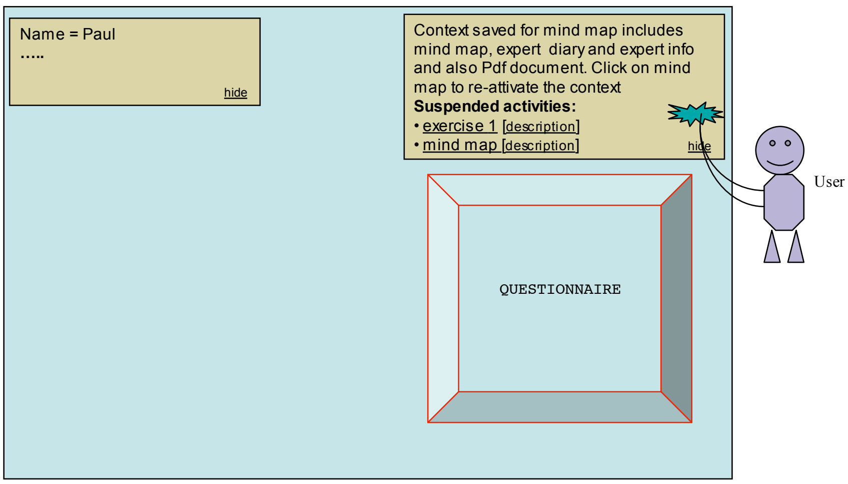
Possible interface level interaction involving scenarios 1 and 2

The agent confirms that the new context for the mind map has been saved and maybe hides/closes the saved windows.



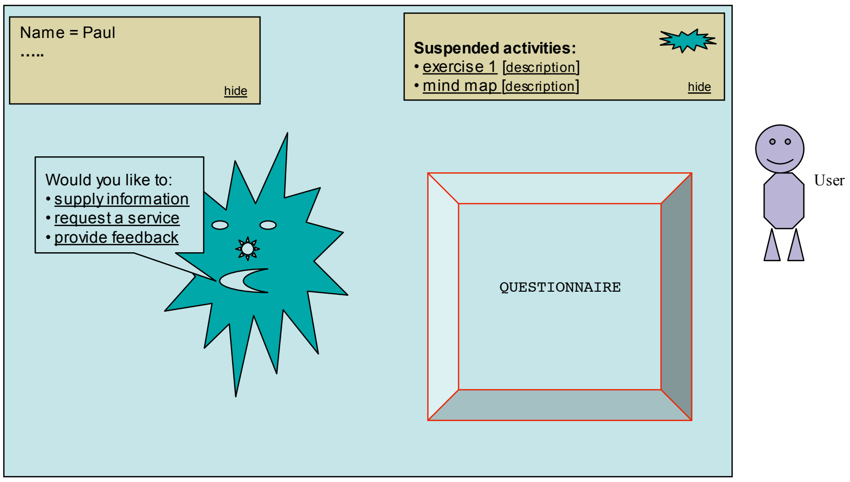
Possible interface level interaction involving scenarios 1 and 2

User calls agent.



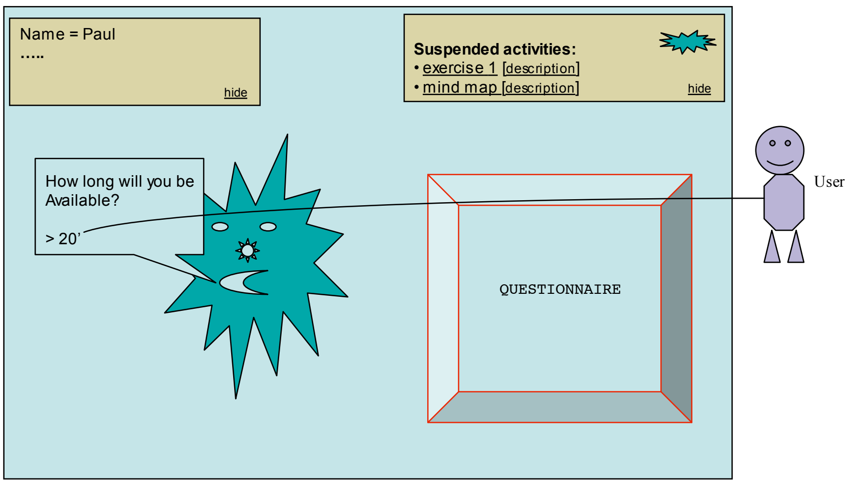
Possible interface level interaction involving scenarios 1 and 2

Who promptly arrives …



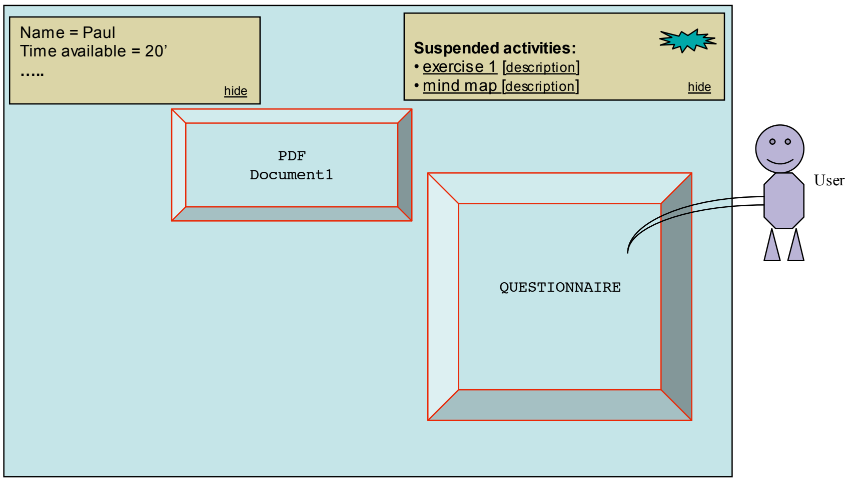
Possible interface level interaction involving scenarios 1 and 2

User indicates that he would like to supply information, and in the contextual menu he indicates that he wants to set the time available; he sets the time to 20'.



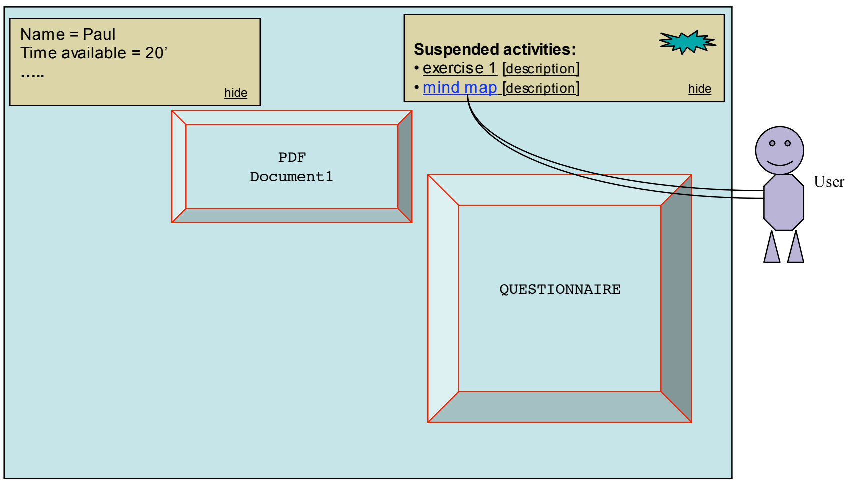
Possible interface level interaction involving scenarios 1 and 2

User model has been updated. The user works at the questionnaire for a while, also opening a new Pdf document ...



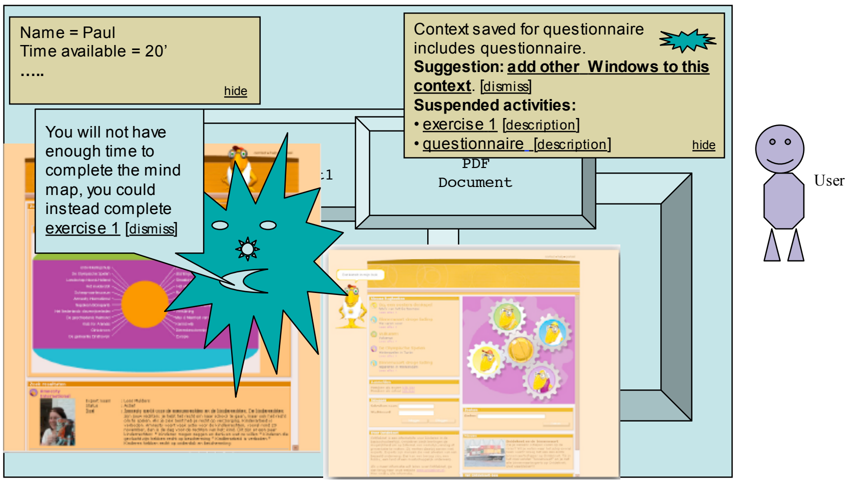
Possible interface level interaction involving scenarios 1 and 2

… then he decides to return to the mind map …



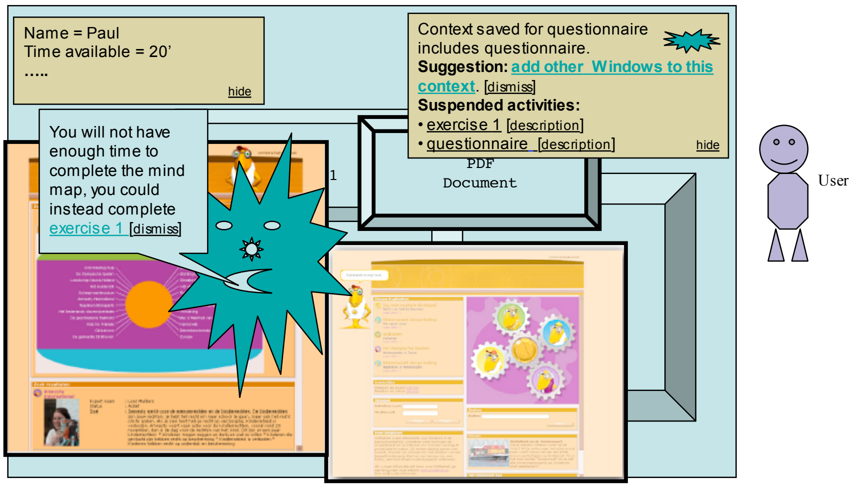
Possible interface level interaction involving scenarios 1 and 2

The mind map context is re-established (mind map is no longer a suspended activity, but the questionnaire is). The agents intervene in two ways: (1) low impact intervention asking the user if he wants to add contextual information to the questionnaire activity; (2) high impact intervention explaining that time available is not sufficient to complete mind map.



Possible interface level interaction involving scenarios 1 and 2

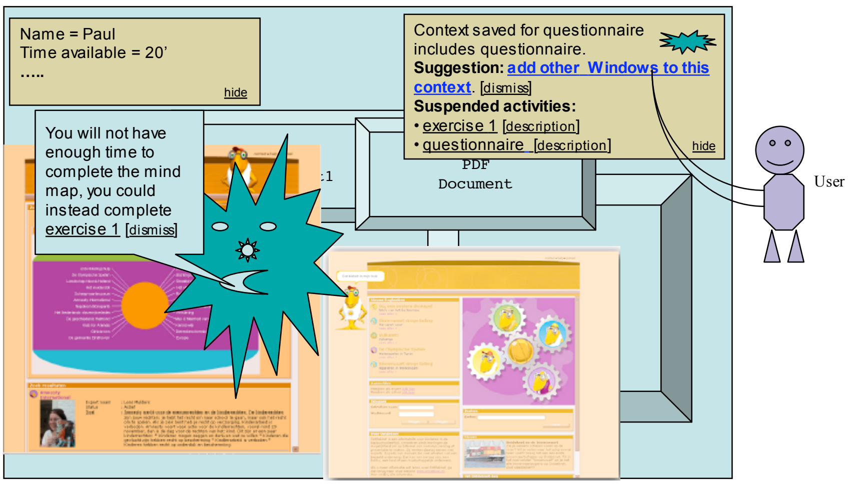
Now things get quite messy (this happens earlier as well but it is not so noticeable!) … what could happen from here? We need a serious evaluation of the possible choices. **I have hypothesizes a few alternatives which you can see by clicking on one of the links below** corresponding to some of the possible user actions. If you just keep clicking on the side you will see all of them.



You have selected to add other windows to the questionnaire context

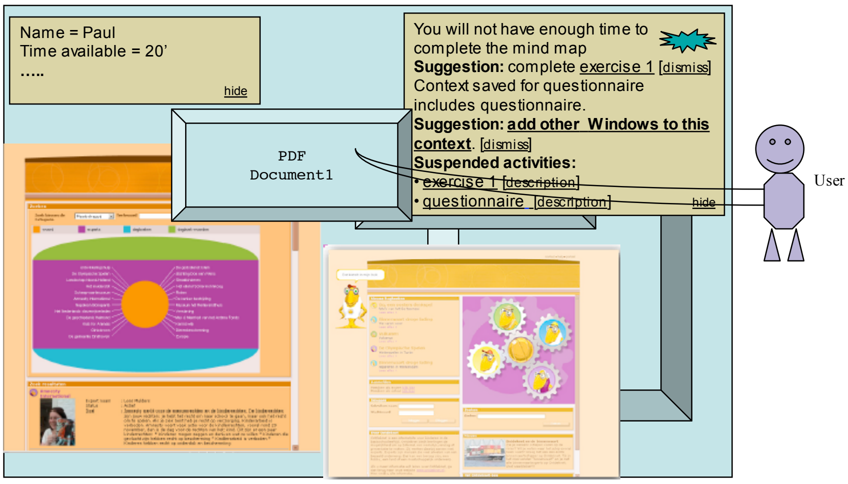
Possible interface level interaction involving scenarios 1 and 2

The user chooses to add more windows to the questionnaire context …



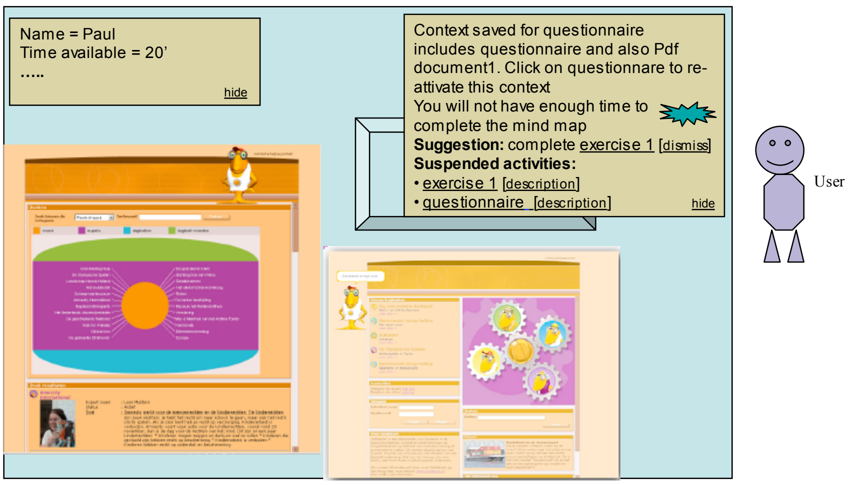
Possible interface level interaction involving scenarios 1 and 2

… and selects the Pdf document1 as a window to add to the context.
Note that, because the agent's high impact intervention's was neither selected nor dismissed, it becomes a low level intervention.



Possible interface level interaction involving scenarios 1 and 2

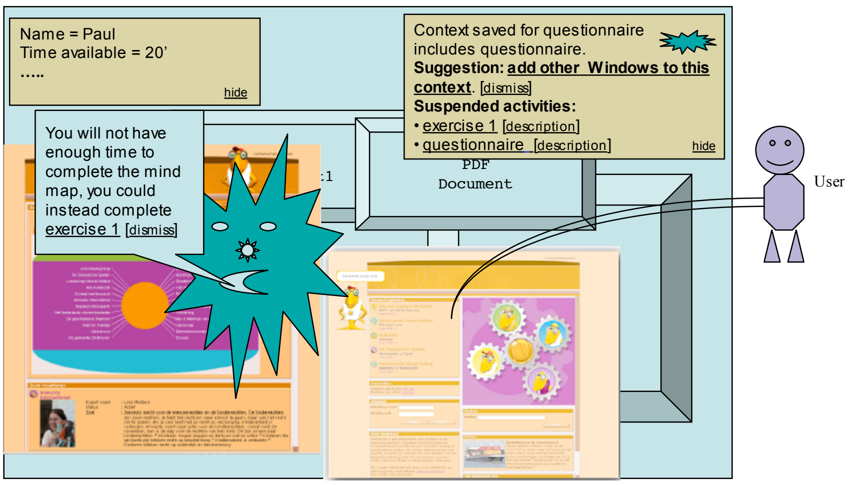
The agent confirms that the new context for the questionnaire has been saved and maybe hides/closes the saved windows.
Now user may either continue working on mind map, or go to exercise 1. **This example finishes here**. <go back to possible user's choices>



You have selected to continue working on the mind map

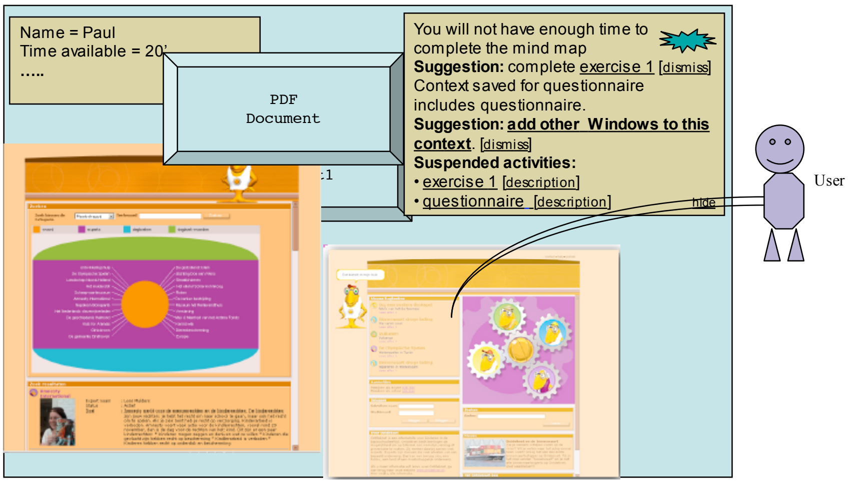
Possible interface level interaction involving scenarios 1 and 2

The user chooses to continue working on the mind map …



Possible interface level interaction involving scenarios 1 and 2

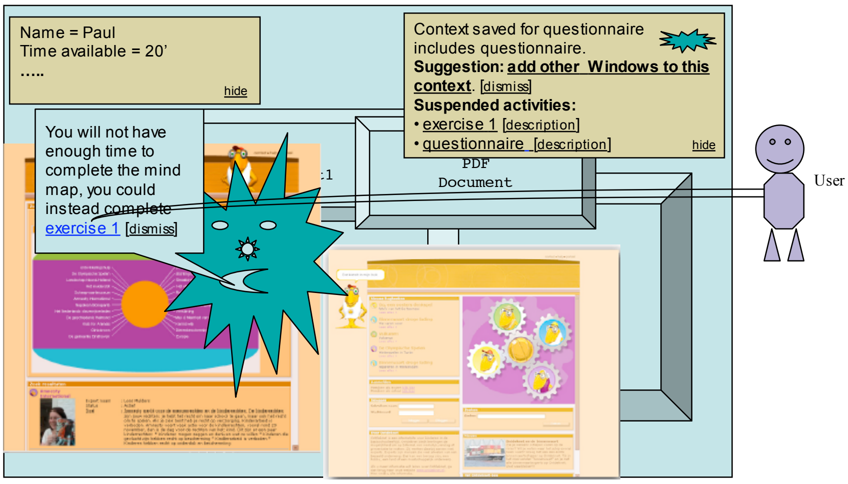
Because the agent’s high impact intervention’s was neither selected nor dismissed, it becomes a low level intervention; also, the only stored window in the questionnaire context is closed/hidden (but not the Pdf document1 window). **This example finishes here**. <go back to possible user’s choices>



You have selected to accept the agent’s suggestion to finish exercise 1

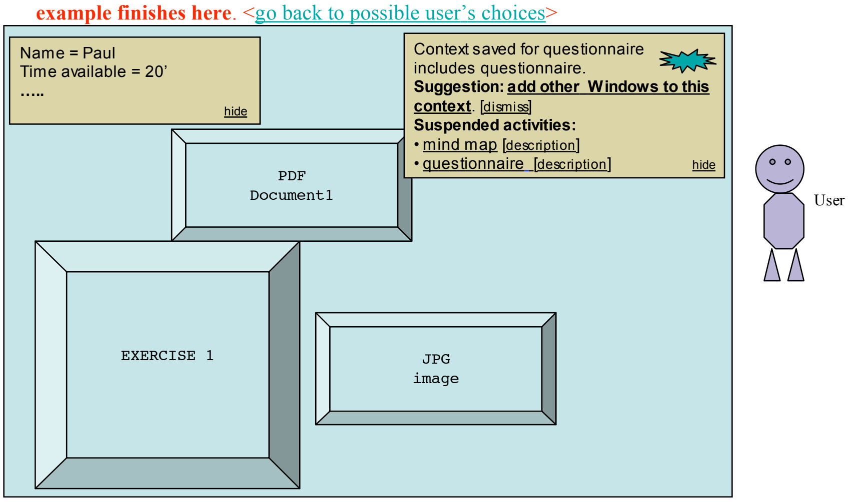
Possible interface level interaction involving scenarios 1 and 2

The user chooses to accept the user suggestion to finish exercise one …



Possible interface level interaction involving scenarios 1 and 2

The mind map context is saved (no new windows were open so the agent does not ask the user to updated the context, and closes/hides it), mind map becomes a suspended activity. No action has been taken on the context of the questionnaire therefore only the questionnaire window is closed (but not the Pdf document 1 window). The context for exercise 1, which includes the exercise 1 window and a JPG image, is restored. **This example finishes here**. <go back to possible user’s choices>



9 Appendix 4 – List of scenarios

Scenario 1: Support to task resumption, restoring task context (I)
Scenario 2: Support to limited time resources allocation
Scenario 3: Notification of external events
Scenario 4: Learning guidance
Scenario 5: User requests notification
Scenario 6: I don't want to do this … bug me no more!
Scenario 7: Re-attracting an idle-user attention
Scenario 8: Re-attracting distracted user's attention
Scenario 9: **Support to task resumption, restoring task context (II)**
Scenario 10: **Restore historical context**
Scenario 11: **Propose task continuation**
Scenario 12: Suggest community relevant resources
Scenario 13: Suggest community relevant tasks
Scenario 14: **Task sequencing**
Scenario 15: Encourage slow user
Scenario 16: Tools for various levels of interruption conspiquity
Scenario 17: Task delegation

Scenario 1: Support to task resumption, restoring task context (I)
The student is working at an assignment. In order to perform this activity he/she has opened the Web page of the course containing the text of the assignment (window 1), a word processor where he/she is typing some text (window 2), as well as a pdf document containing some notes from the professor (window 3). Before completing the assignment, the student switches to another task. Later the student returns to the assignment task; as soon as the student resumes the interrupted task the system proposes to restore the context of the assignment task, as it was left at interruption time, by reopening (or bring to front) the three windows 1, 2, and 3.

Scenario 1: **Support to task resumption, restoring task context (I)**
Applied to: AtgentSchool
The student is building the mind map (current focus) using the expert's introduction diary and personal information, as well as a pdf document opened in an Acrobat window. The student switches to a questionnaire. The AtgentSchool application reports a *start* event for the new activity (questionnaire). The agent saves, possibly with the help of the user, the context of the previous focus (which includes the mind-map window, the diary and information of the expert, and the pdf document window). Later the student returns to the mind-map building activity; the application sends a *resume* event; the agent proposes to restore the saved context.

Scenario 2: Support to limited time resources allocation
The student starts reading the text for a new lecture. The system recognises that a relevant exercise task was previously interrupted (or that the exercise was previously suggested by the application). The agent also evaluates that the exercise task could be completed within the time available to the student whilst reading the text for the new lecture requires longer than the time available to the student. The system suggests working at the exercise.

Scenario 2: Support to limited time resources allocation
Applied to: AtgentSchool
The student starts working at the mind-map (start event). The agents recognise that a relevant exercise task was previously interrupted (or that the exercise was previously suggested by the application). The agent also evaluates that the exercise task could be completed within the time available to the student whilst the mind-map task requires longer than the time available to the student. The agent suggests working at the exercise.

Scenario 3: Notification of external events
The user is performing a task. An email addressed to the user (or other notification event), is received. The system recognises that the message is of average importance (e.g. the sender is listed in the user social network, and the subject is relevant to one of the interrupted tasks) however the system also recognises that the current task is urgent and it requires a heavy workload. The system decides to delay

notifying the user about the message until the occurrence of a breakpoint in the task execution (e.g. the user completes the current activity, or starts a new activity).

Scenario 3: Notification of external events
Applied to: AtgentSchool, AtgentNet
The user is performing a task (e.g. user is working at an assignment in the AtgentSchool application, the user is browsing a space in the AtgentNet application). An email addressed to the user (or other notification event), is received by the application. The application originates a *new information available* event. The agents recognise that the message is of average importance (e.g. the sender is listed in the user social network, and the subject is relevant to one of the interrupted tasks) however the agent also recognises that the current task is urgent and it requires a heavy workload. The agents decide to delay notifying the user about the message until the occurrence of a breakpoint in the task execution (marked by a new *user-application*, or *user* event).

Scenario 4: Learning guidance
The user is reading some information and the application evaluates that the user should also read another document that he/she has not yet explored. The system evaluates the best manner to propose the new focus (on the basis of the user's current and past activity) and makes the suggestion to the user. The user disregards this suggestion (without dismissing it). The system saves the proposed focus to be able to propose it later.

Scenario 4: Learning guidance
Applied to: AtgentNet
The user is visiting one of the platform's *knowledge area* and the application evaluates that the user should also visit another knowledge area, which he/she has not explored. The application generates a *propose focus* event. The agent evaluates the best manner to propose the new focus (on the basis of the current and proposed foci characteristics) and makes the suggestion to the user. The user disregards this suggestion (without dismissing it). The agents save the proposed focus to be able to propose it later.

Scenario 5: User requests notification
The student requests to be notified immediately and with confirmation, about any message coming from a given sender. Upon reception of the email message the system recognises that the conditions for notification are verified, consequently it notifies the user immediately (as requested). Since the user indicated that the notification is *with confirmation*, the notification is repeated at successive breakpoints until the user acknowledges it.

Scenario 5: User requests notification
Applied to: AtgentSchool, AtgentNet
The user requests to be notified immediately and with confirmation, about any message coming from a given sender (*notify-me*). The application, upon reception of email messages, notifies the agents (*new-focus*). The agents recognise that the user wants to be notified about the email. The agents notify the user immediately (as indicated by the *notify-me* event). Because the user indicated that the notification is *with confirmation*, the notification is repeated at successive breakpoints until the user acknowledges it.

Scenario 6: I don't want to do this … bug me no more!
The system proposes to perform a certain task; the user dismisses the proposal. The system will not propose the task again unless the application requires it one more time, in which case the task will be proposed the intervention with further motivation. May ask for reasons for dismissal to the user (e.g. obsolete, too busy, etc.)

Scenario 6: I don't want to do this … bug me no more!
Applied to: AtgentSchool
A child has logged in the AtgentSchool application and is expected to complete the *introduction* activity. The child is new to the activity (he/she has never completed the *introduction* before), has been rated by the teacher as a weak student, has been inactive for a few minutes, and has not reached the *introduction* screen yet. The agents propose some navigational help explaining how to reach the *introduction screen* (e.g. " By clicking on the top left button you will reach the *introduction* screen"). The child dismisses the suggestion. Because the intervention has been dismissed, the Agents will not propose this type of

intervention again unless the application requires it, in which case the task will be proposed with further motivation (e.g. "Before you start working at the mind map you must introduce yourself; it looks like you are having troubles reaching the correct screen. By clicking on the top left button you will reach the *introduction* screen").

Scenario 7: Re-attracting an idle-user attention
The student has started an activity requiring that he/she supplies some input. The student does not provide input for longer than the *maximum input inactivity time* for the task. The system evaluates whether the task being performed is still the best-suited one for the user; it verifies whether the learner is busy with offline activities. Following these evaluations the system may propose to the user: (1) to continue the task, possibly by providing motivation for the task; (2) to receive help on the task; (3) to switch to another relevant task (if available).

Scenario 7: Re-attracting an idle-user attention
Applied to: AtgentSchool
The student has started browsing the expert's information (*start event*). The student does not provide input (*idle input*) for longer than the time indicated as the maximum input inactivity for the task. The agents evaluate if the task being performed is still the best-suited one for the user. The agents consult the user's agenda to verify whether he/she is busy with offline activities. The agents propose to the user to either: (1) to continue the task, possibly by providing motivation for the task; (2) to receive help on the task; (3) to switch to another relevant task (if available).

Scenario 7a: Re-attracting an idle-user attention (a)
The student initiates a task that he/she has never performed before. The student does not provide input for longer than the time indicated as the *maximum input inactivity time* for the task. The system proposes to the student to focus on a support task (e.g. explanation, help) for the task just initiated by the user.

Scenario 7a: Re-attracting an idle-user attention (a)
Applied to: AtgentSchool
The student works at the introduction (*start event*). He/she has never performed an introduction before. The student does not provide input (*idle input*) for longer than the time indicated as the *maximum input inactivity* for the task. The agents propose that the application should provide support for the introduction task. This support may depend, amongst others, on the input already supplied by the student.

Scenario 8: Re-attracting distracted user's attention
The user is active in an application that is not Atgentive enabled as a consequence Atgentive cannot assess whether the user's current focus is more "important" than any of the foci associated to Atgentive enabled applications and doesn't interrupt the user. However, being able to capture window activities such as copy and paste between windows, or frequent windows switches between an Atgentive-application and an "unknown" application, may allow the system to infer which "unknown" windows are part of the context for the current task and therefore make more informed decisions about the user activity.

Scenario 8: Re-attracting distracted user's attention
Applied to: AtgentNet
The user is working at a high priority task on the platform: writing a posting that is due in a few hours. The tracking devices recognise that the user is frequently switching between the platform's window for the post-writing and the window of a document D in a word processor (not Atgentive enabled). The agents tentatively associate the word processing window to the context of the post-writing task. Another tracking device reports an *idle input event* on the post-writing focus. Although this event would normally give rise to an agents' intervention to re-attract the user's attention to the post-writing task, the agents recognise that the user is active in the word processor window for document D. Since this window is associated to the context of the post-writing task, the agents assume that the user is working at the task in another application window and do not intervene.

Scenario 9: **Support to task resumption, restoring task context (II)**
While browsing a document A, the learner has opened several windows; he/she accesses a new document B; the system proposes to the user to select the windows associated to the interrupted browsing activity on document A, in order to save the context of this activity. Later the user re-accesses document A, the

system verifies whether all the windows in the context are already open. If not, it proposes to restore (one of) the saved environment(s) associated to the task of reading document A. The intervention modality will depend, amongst others, on how long the task has been idle.

Scenario 9: Restoring context II
Applied to: AtgentNet
While browsing a knowledge area A, the learner has opened several windows; the user enters a new knowledge area B (*start* event); the agent proposes to the user to select the windows associated to the interrupted browsing activity on A, in order to save the context of that activity. Later the user re-enters the knowledge area A (*start* or *resume* event), the agent verifies whether all the windows in the context are already open. If not, it proposes to restore (one of) the saved environment(s) associated to the task of browsing the knowledge area A. The intervention modality will depend, amongst others, on how long the task has been idle.

Scenario 10: **Restore historical context**
After replying to an email, and reading a document, the user is interrupted while writing a further email. When resuming this last task the system reminds the user that the last actions performed before the interruption consisted in replying to the email and reading the document.

Scenario 10: **Restore historical context**
Applied to: AtgentNet
The system will keep track of the sequence in which the user opens KAs (Knowledge Assests). For every KA, a 'list' will be held of the KAs that were selected immediately both before and afterwards (I will refer to each of these as a "contextual Knowledge Asset"—cKA).
When a user selects a KA the system will look at the last time they opened the same KA and offer the user the n (number to be determined) cKAs which (s)he had previously selected immediately before and after the original KA.
Note that for this scenario a conversation in the Chat window will count as a cKA and the contents may be displayed in a new window as if it were a 'normal' cKA document. (This is because the user may have discussed the current KA with others when it was last in use).
To reduce the cost of interruption, the user will be offered the additional documents (cKAs) only immediately upon selection of a KA. While the user may select one of the proffered cKAs (which will each open in an additional new window), no action need be taken by the user if they so choose.
Once a KA has been selected n times without accepting the contextual KAs the agent will stop offering cKAs for that particular KA (but the user may ask for contextual KAs at any time)..

Scenario 11: **Propose task continuation**
After N observations the user has executed a certain task X after – or interleaved to – a task Y. The user is now focusing again on task Y, once the task is completed the system proposes to continue with task X.

Scenario 11: **Propose task continuation**
Applied to: AtgentNet
After 10 observations the user has looked at the platform's action-log immediately after reading all new messages on the platform 8 times out of 10. The user is now focusing again on the new messages, once this task is completed the agents proposes to continue the activity by looking at the platform's action-log.

Scenario 12: Suggest community relevant resources
As the learner accesses an online resource, say R1, the system offers a set of "related resources". These related resources correspond to those most frequently selected, by all users, immediately both before and after R1. While the user may select one of the proffered related resources, no action need be taken by the user if they so choose.
When a resource is reopened, (i.e. after the first time for that user) the user will be offered the related resources, as described above, AND any related resource accepted previously.

Scenario 12: Suggest community relevant resources
Applied to: AtgentNet
The system keeps track of the sequence in which all users open Knowledge Assests (KAs) in the platform. For every KA, a 'league table' is maintained of the KAs most frequently selected immediately both before and after the main KA (we will refer to each of these as a "related Knowledge Asset - rKA).
When a user selects a KA he/she will be offered the n (number to be determined) rKAs most likely to be of relevance in understanding the KA they chose (i.e. most temporally related).
To reduce the cost of interruption, the user will be offered the additional documents (rKAs) immediately upon selection of a KA. While the user may select one of the proffered rKAs (which will each open in an additional new window), no action need be taken by the user if he/she so choose.
When a KA is reopened, (i.e. after the first time for that user) the user will be offered the most frequently selected rKAs, as described above, AND any rKAs they accepted previous times for the current KA (if they do not now appear as the top n entries in the 'league table').
Once a KA has been selected n times without accepting the related KAs the agent will stop offering rKAs for that particular KA (but the user may ask for related KAs for that KA).

Scenario 13: Suggest community relevant tasks
If a sequence of N events E1 … En generated by this user matches (the event is the same and the task is the same) the beginning of a sequence of M (M>N) events of other users B1, …, Bn, Bn+1, …, Bm, then the task contained in the N+1 event of the sequence (Bn+1) is proposed to this user.

Scenario 13: Suggest community relevant tasks
Applied to: AtgentNet
The sequence of foci <"read D1 on the platform", "read D2 on the platform"> performed by the current users matches the beginning of the sequence <"read D1 on the platform", "read D2 on the platform", "reply to posting D3"> of 5 out of 6 other members of the community. The agents proposed to this user to continue his/her activity by performing "reply to posting D3".

Scenario 14: **Task sequencing**
The learner has completed a task T1 that *must be followed by* task T2. Upon completion of T1, the learner is informed that the next task to be completed is T2. Similarly, other constraints may be defined on tasks sequences, for example, that a task T1 must be completed before initiating task T2.

Scenario 14: **Support to task continuation: required sequence**
Applied to: AtgentSchool
The application has informed Atgentive that the task *login* must be followed by the task *introduction*. Once the learner has completed a task *login* he/she is informed that the next task to be completed is the *introduction*. Similarly, other constraints may be defined on tasks sequences, for example, the *introduction* must be completed before *contacting the expert*.

Scenario 15: Encourage slow user
The student initiates a task that he/she has never performed before. The student provides input with a frequency lower than the *minimum input frequency* for the task. The system supplies some encouragement and perhaps some simple explanations. When the learner's input frequency increases, the system gives a positive feedback.

Scenario 15: Encourage slow user
Applied to: AtgentSchool
The student starts with the *introduction* task that he/she has never performed before. In the *start* event the application has indicated a *minimum input frequency* for the task. The student provides input with a frequency lower than the *minimum input frequency* (*low input frequency event*). The system supplies some encouragement and perhaps some simple explanations relative to the *introduction* task. When the learner's input frequency increases, the system gives a positive feedback.

Scenario 16: Tools for various levels of interruption conspiquity
The learner must be notified about new documents available for his/her course. This information is defined as having a *low urgency* and a *high content* level. The system will pass on this information as an *email*.
Later, the learner must be notified about a real time chat meeting with the teacher that will take place in 5 minutes. This information is defined as having a *high urgency* and a *low content* level, and an *action tracking* on the "user connecting in the chat meeting". The learner is notified about the chat event by an instant message.
Later yet, if the user has not connected in the chat event, he is notified, with a further instant message, about the number of participants already in the chat meeting.

Scenario 16: Tools for various levels of interruption conspiquity
Applied to: AtgentNet
For each entry on the platform, the AtgentNet application generates a *new information available* event indicating that this is a "new platform entry", that the urgency is *low*, and that the *content level* is *high*.
The user has indicated, with a *set interruption frequency* event, that the maximum interruption frequency for the "new platform entry" information is *weekly,* and that the interruption modality should be by email.
The agents collect all "new platform entry" information and inform the user with a weekly email summarizing the activities of the last period (such as the number of messages that have been posted, the title of the messages, and some indicators of the activity of the community).
Later, the AtgentNet application generates a *new information available* event indicating that:

1) this is a "new chat meeting",
2) that the urgency is *high* (the meeting will take place in five minutes),
3) that the *content level* is *low*, and
4) that the application requires notification if the user does not connect to the chat within 5 minutes.

The agents notify the user about the chat event with an instant message.
Since the user does not login in the chat within 5 minutes, the agents notify the application.
The application generates a further *new information available* event that results in the user receiving a further instant message, about the number of participants already in the chat meeting.

Scenario 17: Task delegation
In a virtual learning community, the community organizer creates a message to be sent to the community, he/she can also indicate presentation style and media, the time of delivery, as well as the operations that should take place after delivery (for instance the message may be archived after it has been read by all recipients, or a reminder may be sent to recipients who did not reply). The system will take charge of completing *after delivery* actions.

10 References

Abowd, G. D., Mynatt, E. D., & Rodden, T. (2002). The human experience. IEEE Pervasive Computing, 1(1), 48-57.

Adamczyk, P. D., & Bailey, B. P. (2004). If not now, when? The effects of interruption at different moments within task execution. Proceedings Human Factors in Computing Systems: CHI'04, New York, 271-278.

Adamczyk, P. D., & Bailey, B. P. (2005, September 26-27). A method and system for intelligent interruption management. Proceedings 4th International Workshop on Task Models and Diagrams for User Interface Design, Gdansk, Poland.

Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE Transactions on Knowledge and Data Engineering, 17(6), 734 - 749.

Angehrn, A. (1993). Computers that criticize you: Stimulus based decision support systems. Interfaces, 23(3), 3-16.

Arroyo, E., & Selker, T. (2003). Self-adaptive multimodal-interruption interfaces. Proceedings International Conference on Intelligent User Interfaces (IUI'03), 6-11.

Bailey, B. P., Adamczyk, P., Chang, T. Y., & Chilson, N. A. (2006). A framework for specifying and monitoring user tasks. Computers in Human Behavior, 22(2), 709-732.

Bailey, B. P., & Konstan, J. A. (2006). On the need for attention aware systems: Measuring the effects of interruption on task - performance, error rate, and affective state. Computers in Human Behavior, 22(4), 685-708.

Bailey, B. P., Konstan, J. A., & Carlis, J. V. (2001). The effects of interruptions on task performance, annoyance, and anxiety in the user interface. Proceedings INTERACT '01, 593-601.

Bartram, L., Ware, C., & Calvert, T. (2003). Moticons: Detection, distraction and task. International Journal of Human-Computer Studies, 58(5), 515-545.

Bearne, M., Jones, S., & Sapsford-Francis, J. (1994, October 15-20 1994). Towards usability guidelines for multimedia systems. Proceedings second ACM international conference on Multimedia, San Francisco, California, United States, 105 - 110.

Beier, B., & Vaughan, M. W. (2003). Web usability: The bull's-eye: A framework for web application user interface design guidelines. Proceedings SIGCHI conference on Human factors in computing systems.

Bier, E. A., Stone, M. C., Pier, K., Buxton, W., & DeRose, T. D. (1993, September). Toolglass and magic lenses: The see-through interface. Proceedings 20th annual conference on Computer graphics and interactive techniques, 73-80.

Bohnenberger, T., Brandherm, B., Grossmann-Hutter, B., Heckmann, D., & Wittig, F. (2002). Empirically grounded decision-theoretic adaptation to situation-dependent resource limitations. Künstliche Intelligenz., 3, 10-16.

Brush, B., Bargeron, D., Gupta, A., & Grudin, J. (2001). Notification for shared annotation of digital documents (Technical Report No. MSR-TR-2001-87): Microsoft Research Microsoft Corporation.

Carroll, J. M., Neale, D. C., Isenhour, P. L., Rosson, M. B., & McCrickard, D. S. (2003). Notification and awareness: Synchronizing task-oriented collaborative activity. International Journal of Human-Computer Studies, 58(5), 605-632.

Chalmers, M. (2000). When cookies aren't enough: Tracking and enriching web activity with recer. In Preferred placement: Knowledge politics on the web (pp. 99-102). Amsterdam: Jan van Eyck Academie Editions.

Clauzel, D., Roda, C., & Stojanov, G. (2006). Tracking task context to support resumption (position paper for 2006 workshop on computer assisted recording, pre-processing, and analysis of user interaction data). Proceedings HCI 2006 - Engage, London, UK.

Cutrell, E., Czerwinski, M., & Horvitz, E. (2001, July 2001). Notification, disruption, and memory: Effects of messaging interruptions on memory and performance. Proceedings Interact 2001 IFIP Conference on Human-Computer Interaction, Tokyo, Japan.

Czerwinski, M., Cutrell, E., & Horvitz, E. (2000, 5-8 September 2000). Instant messaging: Effects of relevance and time. Proceedings HCI 2000 - 14th British HCI group Annual Conference, University of Sunderland, 71–76.

Czerwinski, M., & Horvitz, E. (2002). Memory of daily computing events. Proceedings HCI 2002 People and Computers XVI, 230-245.

Czerwinski, M., Horvitz, E., & Wilhite, S. (2004). A diary study of task switching and interruptions. Proceedings Proceedings of the SIGCHI conference on Human factors in computing systems, Vienna, Austria, 175-182.
Egeth, H., & Yantis, S. (1997). Visual attention: Control, representation, and time course. Annual Review of Psychology, 48, 269-297.
Erickson, T., Halverson, C., Kellogg, W. A., Laff, M., & Wolf, T. (2002). Social translucence: designing social infrastructures that make collective activity visible. Communications of the ACM, 45(4), 40 - 44.
Fogarty, J., Hudson, S. E., Atkeson, C. G., Avrahami, D., Forlizzi, J., Kiesler, S., Lee, J. C., & Yang, J. (2005). Predicting human interruptibility with sensors. ACM Transactions on Computer-Human Interaction, 12(1), 119–146.
Fogarty, J., Hudson, S. E., & Lai, J. (2004). Examining the robustness of sensor-based statistical models of human interruptibility. Proceedings Proceedings of the SIGCHI conference on Human factors in computing systems, Vienna, Austria, 207-214.
Fogarty, J., Ko, A. J., Aung, H. H., Golden, E., Tang, K. P., & Hudson, S. E. (2005). Examining task engagement in sensor-based statistical models of human interruptibility. Proceedings Proceedings of the SIGCHI conference on Human factors in computing systems, Portland, Oregon, USA, 331-340.
Fogarty, J., Lai, J., & Christensen, J. (2004). Presence versus availability: The design and evaluation of a context-aware communication client. International Journal of Human-Computer Studies, 61(3), 299-317.
Franke, J. L., Daniels, J. J., & McFarlane, D. C. (2002). Recovering context after interruption. Proceedings 24th Annual Meeting of the Cognitive Science Society (CogSci 2002), 310-315.
Garner, R. (1987). Metacognition and reading comprehension. Norwood, NJ: Ablex.
Gievska, S., Lindeman, R., & Sibert, J. (2005). Examining the qualitative gains of mediating human interruptions during hci. Proceedings 11th International Conference on Human-Computer Interaction, Las Vegas, Nevada.
Gonzalez, V. M., & Mark, G. (2004). "constant, constant, multi-tasking craziness": Managing multiple working spheres. Proceedings Proceedings of the SIGCHI conference on Human factors in computing systems, Vienna, Austria, 113-120.
Healey, C. G. (2005, December). Perception in visualization. Retrieved 30 June, 2006, from http://www.csc.ncsu.edu/faculty/healey/PP/
Ho, C. Y., Nikolic, M. I., Waters, M. J., & Sarter, N. B. (2004). Not now! Supporting interruption management by indicating the modality and urgency of pending tasks. Human Factors, 46(3), 399-409.
Ho, J., & Intille, S. S. (2005). Using context-aware computing to reduce the perceived burden of interruptions from mobile devices. Proceedings Proceedings of the SIGCHI conference on Human factors in computing systems, Portland, Oregon, USA, 909-918.
Horvitz, E., & Apacible, J. (2003). Learning and reasoning about interruption. Proceedings 5th International Conference On Multimodal Interfaces, Vancouver, British Columbia, Canada, 20 - 27.
Horvitz, E., Kadie, C., Paek, T., & Hovel, D. (2003). Models of attention in computing and communication: From principles to applications. Communications of the ACM, 46(3), 52-59.
Horvitz, E., Koch, P., & Apacible, J. (2004, November 2004). Busybody: Creating and fielding personalized models of the cost of interruption. Proceedings Conference on Computer Supported Cooperative Work, Chicago, Illinois.
Horvitz, E., Koch, P., Kadie, C. M., & Jacobs, A. (2002, July 2002). Coordinate: Probabilistic forecasting of presence and availability. Proceedings Eighteenth Conference on Uncertainty and Artificial Intelligence, Edmonton, Alberta, 224-233.
Hudson, J. M., Christensen, J., Kellogg, W. A., & Erickson, T. (2002, April 20-25). "i'd be overwhelmed, but it's just one more thing to do": availability and interruption in research management. Proceedings SIGCHI conference on Human factors in computing systems, Minneapolis, Minnesota, USA, 97 - 104.
Iqbal, S. T., Adamczyk, P. D., Zheng, X. S., & Bailey, B. P. (2005). Towards an index of opportunity: Understanding changes in mental workload during task execution. Proceedings Proceedings of the SIGCHI conference on Human factors in computing systems, Portland, Oregon, USA, 311-320.
Ivory, M. Y., & Megraw, R. (2005). Evolution of web site design patterns. ACM Transactions on Information Systems (TOIS), 23(4).

Jameson, A., Schafer, R., Weis, T., Berthold, A., & Weyrath, T. (1999). Making systems sensitive to the user's time and working memory constraints. In Proceedings of the 4th international conference on intelligent user interfaces (pp. 79-86). Los Angeles, California, United States: ACM Press.
Jersild, A. (1927). Mental set and shift. Archives of Psychology (Whole No. 89) - Referenced in Rubinstein, Meyer, and Evans 2001.
Kvavilashvili, L., Messer, D. J., & Ebdon, P. (2001). Prospective memory in children: The effects of age and task interruption. Developmental Psychology, 37(3), 418-430.
Lester, J., Converse, S., Kahler, S., Barlow, T., Stone, B., & Bhogal, R. (1997). The persona effect: Affective impact of animated pedagogical agents. Proceedings CHI '97, 359-366.
Mark, G., Gonzalez, V. M., & Harris, J. (2005). No task left behind? Examining the nature of fragmented work. Proceedings Proceedings of the SIGCHI conference on Human factors in computing systems, Portland, Oregon, USA, 321-330.
Marsh, R. L., Hicks, J. L., & Bryan, E. S. (1999). The activation of un-related and canceled intentions. Memory & Cognition, 27(2), 320-327.
Maslach, C., Schaufeli, W. B., & Leiter, M. P. (2001). Job burnout. Annual Review of Psychology, 52, 397-422.
McCrickard, D. S., Catrambone, R., Chewar, C. M., & Stasko, J. T. (2003). Establishing tradeoffs that leverage attention for utility: Empirically evaluating information display in notification systems. International Journal of Human-Computer Studies, 58(5), 547-582.
McCrickard, D. S., & Chewar, C. M. (2003). Attuning notification design to user goals and attention costs. Communications of the ACM, 46(3), 67-72.
McFarlane, D. C., & Latorella, K. A. (2002). The scope and importance of human interruption in human-computer interaction design. Human-Computer Interaction, 17(1), 1-62.
Meacham, J. A., & Leiman, B. (1982). Remembering to perform future actions. In U. Neisser (Ed.), Memory observed: Remembering in natural contexts (pp. 327-336). San Francisco: Freeman.
Miller, G. (1956). The magical number seven, plus or minus two. Psychological Review, 63, 81-97.
Nagata, S. F. (2003). Multitasking and interruptions during mobile web tasks. Proceedings 47th Annual Meeting of the Human Factors and Ergonomics Society, 1341-1345.
Nelson, T. (1996). Consciousness and metacognition. American Psychologist, 51(2), 102-116.
O'Conaill, B., & Frohlich, D. (1995, 7-11 May 1995). Timespace in the workplace: Dealing with interruptions. Proceedings CHI '95 Conference Companion, Denver, Colorado, United States, 262-263.
Obermayer, R. W., & Nugent, W. A. (2000). Human-computer interaction for alert warning and attention allocation systems of the multi-modal watchstation. Proceedings SPIE 2000 - The International Society for Optical Engineering.
Picard, R. W. (1997). Affective computing. Cambridge, MA: MIT Press.
Picard, R. W. (2004). Human responses to technology scrutinized. The Washington Post.
Rensink, R. A. (2000). Seeing, sensing, and scrutinizing. Vision Research, 40(10-12), 1469-1487.
Robertson, T. J., Prabhakararao, S., Burnett, M., Cook, C., Ruthruff, J. R., Beckwith, L., & Phalgune, A. (2004). Impact of interruption style on end-user debugging. Proceedings Proceedings of the SIGCHI conference on Human factors in computing systems, Vienna, Austria, 287-294.
Roda, C. (Ed.). (2006). Atgentive (ist-4-027529-stp) deliverable d1.2 - state of the art report.
Roda, C., & Nabeth, T. (2006). The atgentive project: Attentive agents for collaborative learners. Proceedings First European Conference on Technology Enhanced Learning EC-TEL'06, Crete, Greece.
Roda, C., & Thomas, J. (2006). Attention aware systems: Theories, applications, and research agenda. Computers in Human Behavior, 22(4), 557–587.
Rubinstein, J. S., Meyer, D. E., & Evans, J. (2001). Executive control of cognitive processes in task switching. Journal of Experimental Psychology: Human Perception and Performance, 27(4), 763-797.
Rudman, P., & Zajicek, M. (2006 submitted-a). Artificial agents and the art of persuasion. Proceedings IAT 2006 - IEEE/WIC/ACM International Workshop on Communication between Human and Artificial Agents, Hong Kong.
Rudman, P., & Zajicek, M. (2006 submitted-b). Autonomous agent as helper – helpful or annoying? Proceedings IAT 2006 - IEEE/WIC/ACM International Conference on Intelligent Agent Technology, Hong Kong.
Rudman, P., & Zajicek, M. (Eds.). (2006). Atgentive (ist-4-027529-stp) deliverable d4.2 - result of the formative evaluation.

Sellen, A. J., Louie, G., Harris, J. E., & Wilkins, A. J. (1996). What brings intentions to mind? An in situ study of prospective memory. Memory & Cognition, 5(4), 483-507.
Sifry, D., Marks, K., Çelik, T., & Hayes, E. (2006). Attention.Xml. Retrieved 30 June, 2006, from http://developers.technorati.com/wiki/attentionxml
Speier, C., Vessey, I., & Valacich, J. S. (2003). The effects of interruptions, task complexity, and information presentation on computer-supported decision-making performance. Decision Sciences, 34(4), 771-797.
Stojanov, G., & Roda, C. (Eds.). (2006). Atgentive (ist-4-027529-stp) deliverable d1.1 - report on atgentive wp1 workshop on "advanced support for attention in collaborative learning settings".
Toet, A. (2006). Gaze directed displays as an enabling technology for attention aware systems. Computers in Human Behavior, Forthcoming(Special issue on Attention aware systems. Roda & Thomas editors.).
Treisman, A. (1998). Features and objects: The fourteenth bartlett memorial lecture. Quarterly Journal of Experimental Psychology, 40A, 201-237.
Treisman, A., & Gelade, G. (1980). A feature-integration theory of attention. Cognitive Psychology, 12, 97-136.
Whittaker, S., Terveen, L., & Nardi, B. (2000). Let's stop pushing the envelope and start addressing it: A reference task agenda for hci. Human Computer Interaction, 15, 75-106.
Wolfe, J., Klempen, N., & Dahlen, K. (2000). Postattentive vision. Journal of Experimental Psychology: Human Perception & Performance, 26(2), 693-716.
Wolfe, J. M. (2001). The level of attention: Mediating between the stimulus and perception. In L. Harris (Ed.), Levels of perception: A festschrift for ian howard: Springer Verlag; Accessed online (15.2.05) draft of 8/2/01, 12:06 PM at http://search.bwh.harvard.edu/pdf/Festschrift.pdf.
Zijlstra, F. R. H., Roe, R. A., Leonova, A. B., & Krediet, I. (1999). Temporal factors in mental work: Effects of interrupted activities. Journal of Occupational and Organizational Psychology, 72, 163-185.

Extended Table of Contents

Index