Supporting Conceptual Awareness with Pedagogical Agents

Anders Mørch, Silje Jondahl, Jan A. Dolonen

HAL Id: hal-00190528
https://telearn.hal.science/hal-00190528

Submitted on 23 Nov 2007

Supporting Conceptual Awareness with Pedagogical Agents

Anders I. Mørch
Associate Professor
InterMedia, University of Oslo
P.O. Box 1161 Blindern
N-0318 Oslo, Norway
Email: anders.morch@intermedia.uio.no

Silje Jondahl[*]
IT Consultant
The County Governor in Rogaland
P.O Box 210, N-4001 Stavanger, Norway
Email: sjondahl@hotmail.com

Jan A. Dolonen
Web Developer
InterMedia, University of Oslo
P.O. Box 1161 Blindern
N-0318 Oslo, Norway
Email: j.a.dolonen@intermedia.uio.no

---

[*]The CoPAS study presented in this article was performed while the author was a graduate student at the Department of information science, University of Bergen, Norway.

Abstract

This paper describes a series of efforts in building and conceptualizing software agents for distributed collaborative learning. The agents are referred to as pedagogical agents. We have integrated pedagogical agents within two collaborative environments, TeamWave Workplace and Future Learning Environment. The role of agents in these environments differs from past work on software agents in their function as extended awareness mechanisms, focusing on task and concept awareness (conceptual awareness). Our approach is stimulated by Mead's theory of the 'Generalized Other'. The agents collect statistical information on user activity and analyze the information based on principles of collaboration and knowledge building (participation, group interaction, and scientific discourse). Furthermore, the agents define a trajectory in a pedagogical agent design space, which we define in terms of four dimensions: presentation, intervention, task, and pedagogy. We end the paper by comparing our approach with related work.

1. Introduction

Over the past 10 years considerable progress has been made towards developing technology for immersive presence in Human Computer Interaction (HCI) and Computer-Supported Cooperative Work (CSCW), such as virtual reality (Bates, 1992), augmented reality, and awareness (Dourish & Bellotti, 1992). Pedagogical agents extend this to educational settings (Johnson, Rickel, Stiles & Munro, 1998) by being software programs that support human learners' awareness of their own and other learners' actions and activities in a virtual learning environment. In this paper we profile pedagogical agents as a conceptual awareness mechanism that identifies the 'common attitude' and the shared principles constituting a virtual learning environment, which is more than the sum of individual actions and activities.

Awareness is a central concept in CSCW and Computer-Supported Collaborative Learning (CSCL) that needs to be addressed when developing pedagogical agents for CSCL environments. Awareness includes both computational mechanisms and higher-level concepts. The main emphasis of the past work (especially in CSCW) has been on computational mechanisms for synchronous collaboration environments, including multiple cursors and echoes (Beaudouin-Lafon & Karsenty, 1992). However, awareness in asynchronous collaboration environments is important as well, for example supported by shared object interaction histories, structuring of discussion threads, and electronic Post-It notes (Gutwin, Stark & Greenberg, 1995). A commonly used definition of awareness describes it as "*an understanding of the activities of others, which provides a context for your own activity*" (Dourish & Bellotti, 1992). The context can help interpret

others' actions and ensure that individual contributions are relevant to the group's overall activity, and it enables evaluation of individual actions with respect to the group's goals. Gutwin at al. (1995) have suggested four types of awareness that may be of relevance in order to scaffold learning in a CSCL environment: workspace, social, task, and concept. Workspace awareness is the up-to-the-minute knowledge a person holds about the state of others' interactions with the environment and can reduce efforts needed to coordinate common tasks and actions relative to shared objects. Social awareness is awareness of the social interactions within a group, for example, whether or not another person is paying attention to what one is doing. This form of awareness is associated with non-verbal cues and emotional states (such as head nods and facial expressions) and has been approximated by video conferencing systems. Task awareness is awareness of how to complete a common task (such as a learning assignment) as well as understanding the purpose of the task. Finally, concept awareness is awareness of how a particular piece of knowledge fits into the students' existing knowledge. These important aspects of awareness are by no means solved by the current state of the art in awareness technology. We see this as a challenge for our own efforts, which we consider from both the conceptual as well as the technical perspectives. Conceptually, we address it by articulating a design space for pedagogical agents, and technically we have built prototypes of agents that provide learners with awareness of other learners' collaboration and knowledge-building activities in a distributed CSCL setting.

2. Theoretical background

Pedagogical agents for computer-supported collaborative learning build on previous work in groupware, collaborative learning, and software agents. In addition we make use of the notion of the 'Generalized Other' (Mead, 1934; Dodds & Valsiner, 1997) in order to ground our approach theoretically. A brief background of these areas is presented below.

2.1 Groupware and CSCW

CSCW and groupware emerged as established research areas during the mid-1980s by bringing together researchers from a variety of fields, including computer science (Ellis, Gibbs & Rein, 1991), information systems (DeSanctis, 1987), Computer-Mediated Communication (Hiltz & Turoff, 1985), HCI (Bannon, 1986), social psychology (Kiesler, Siegel & McGuire, 1984), ethnography (Hughes, Randall & Shapiro, 1992), and organizational science (Huber, 1982). A shared interest has been to study how collaboration technology impacts work environments and how this technology can best be developed (Grudin, 1994). The technology in CSCW is often referred to as groupware (Johnson-Lenz & Johnson-Lenz, 1982), but it encompasses a wide range of systems, including mail systems, news groups and bulletin boards, information sharing systems (such as discussion forums), and advanced applications for facilitating synchronous communication such as group editors, multimedia spaces, and video conferencing systems. During the past few years the functionality of this technology has gradually matured and stabilized. As a result many of today's desktop computers provide built-in groupware tools that cater to a host of cooperative work arrangements and there are emerging standards for information sharing, concurrency control, and awareness technology.

2.2 Computer-Supported Collaborative learning

Computer-Supported Collaborative Learning (CSCL) a new field that emerged during the last decade and concerns educational technology and its role as mediator of activity within a collaborative setting of instruction and learning (e.g. Dillenbourg, Baker, Blaye & O'Malley, 1996; Koschmann, 1996). Groupware is often used as the technological platform for achieving this. CSCL as a field is complex and multifaceted and borrows ideas from the social sciences, particularly from sociology, anthropology, linguistics, and communication (Koschmann, 1996). Researchers and educators in these areas have questioned and rejected earlier approaches to computer-based instruction because, in those areas, learning is perceived as belonging primarily to the realm of cognitive psychology. Instead the focus in CSCL has shifted to the socially constructed properties of interaction (Dillenbourg, et al., 1996). Knowledge, from this perspective, is seen as a human construction elaborated through communication and collaboration with peers, mediated by social and cultural artifacts (e.g. language, technology), implying that learning and knowledge building first of all occur on interpersonal grounds within a community of learners before occurring on the intrapersonal realm of the individual learner (Vygotsky, 1978).

2.3 Mead's Theory of the 'Generalized Other'

Within the socio-cultural perspective on CSCL Vygotsky is a frequently cited reference when emphasizing the role of social interaction in learning processes (e.g. Koschmann, 1996). A predecessor of Vygotsky (1978) by some twenty-five years, George Herbert

Mead (1934) made important contributions to the study of human development as it arises in the context of social interaction with significant others (Mead, 1934; Vygotsky, 1978), thus providing additional theoretical motivation for CSCL, especially for the design of virtual learning environments. Mead proposed the '*game*' as model for the social aspects of personal development. He suggested that individual participants learn by internalizing the roles of other players and the rules of the game. For a newcomer this means to engage in a social learning situation by imitating and responding to unfamiliar gestures, gradually adopting the attitudes held by the senior players, and eventually even modifying them. The shared representation of the set of common attitudes is called the '*Generalized Other*'. This representation is different from the sum of individual participants' actions and activities in the game, or in Mead's own words: "*In the game, the individual is required to internalize, not merely the character of a single and specific other, but the roles of all others who are involved with him in the game. He must, moreover, comprehend the rules of the game that condition the various roles. This configuration of roles-organized-according-to-rules brings the attitudes of all participants together to form a symbolized unity: this unity is the 'generalized other'. The generalized other is 'an organized and generalized attitude' with reference to which the individual defines her own conduct.*" (Mead, 1934, p. 151).

The notion of the '*Generalized Other*' serves as a theoretical motivation for incorporating conceptual awareness in distributed learning environments. A distributed learning environment has a set of '*rules*' for how to interact. Furthermore, these rules are not straightforward for most participants. The '*players*' need to learn at least one role in order

to successfully participate (i.e. to be a collaborator). When this set of common attitudes is internalized and shared among the other users, it may improve participation and collaboration, a claim we address in this article. Finally, if software agents are allowed to reason with these representations, conceptual awareness can be implemented as a computational mechanism. For example, by monitoring the shared state of a groupware system, the agents can automatically generate representations of who is logged on, who communicates with whom, what objects they act upon, how much of a task has been completed, etc. These findings can be organized in various meaningful formats depending on the target audience (e.g. students, instructors). A working definition of pedagogical agents from this perspective, building on the definition of awareness provided by Dourish & Bellotti (1992), would be: An understanding of the '*generalized activity'* of others, which provides a context for ones own activity.

2.4 Software agents

2.4.1 General characteristics

The first idea of a software agent goes back to the mid-1950s when John McCarthy proposed a system called the *"advice taker"* (McCarthy, 1959). These first agents were a collection of multiple, goal-seeking entities that compete and cooperate inside the computer to produce intelligent behavior based on a high-level goal. When the system became stuck it could ask a human operator for advice and continue with its operations.

With the emergence of personal computing and graphical user interfaces in the mid-1980s, new ways of using the computer reversed the roles of humans and software

agents. The agents became an assistant to users rather than the other way around. Apple Computer presented one of the first scenarios of this type of human-computer interaction in a 1987 promotion video (Apple Computer, 1987; Scully, 1989). The featured system, Knowledge Navigator, had a "*butler*" agent that helped a domain-expert user search for information in a large information space. This scenario foreshadowed the Web as the "*new world*" where software agents would conduct their business. Therefore, during the past 20 years, the world of agents has expanded from the computer (Kay, 1984) into the network (Maes, 1994) and onto the World Wide Web (e.g. Liu, Zhong, Yao & Ras, 2003).

Nwana (1996) has proposed a typology of agents (Collaborative Learning Agents, Smart Agents, Interface Agents and Collaborative Agents) that are characterized by their combination of two or more of the following three principles: autonomous action, cooperation, and learning (Figure 1). Autonomous action refers to the principle that agents can operate without human interference, cooperation is the ability an agent has to communicate with the user or with other agents, and agent learning refers to an agent's capability of changing its behavior over time as a result of past cooperation with the goal of improving performance. These four types are neither mutually exclusive nor do they represent an exhaustive list of all possible variations of agents, but rather represent the plurality in the agent literature.

Insert Figure 1 about here

Based on a comprehensive literature survey, Baggetun, Dolonen & Dragsnes (2001) identified the following attributes of software agents:

- Autonomous: ability to operate without human interference;

- Communicative: ability to communicate with other agents and users;

- Reactive: monitoring and reacting to the state of its environment; and

- Adaptive: adaptable internal representation of its working environment.

According to Nwana's (1996) typology, our pedagogical agents can be described as autonomous agents reacting to changes in their environment, communicating in rudimentary ways with other agents, communicating directly with the users, and being adaptable by end users. This type of agent is also called an '*interface agent*.' According to Lieberman and Selker (2003) for an agent to be considered an interface agent it should communicate directly with the user through the system's input and output devices. Furthermore, it should be able to monitor actions performed by a user in a direct manipulation interface and provide feedback to users by invoking commands in the system.

2.4.2 Agents in education

The first generation of agents in educational technology was associated with intelligent tutoring systems (ITS) (Anderson, Boyle, Farrell & Reiser, 1987), computer-based coaches (Burton & Brown, 1982; Selker, 1994), and critic systems (Fisher, Lemke, Mastaglio & Mørch, 1991), although the term agent has not always been used with these

initiatives. Tutors share with agents aspects of the original definition proposed by McCarthy, i.e. top-down execution from on a well-defined goal to local actions and operations. Coaches can be positioned in-between tutors and critics along a continuum from formal to informal intelligent support systems. Early coaches have served as the model for both tutors and critics. A critic system resembles a computer-based coach, but it operates in the context of a domain-specific micro-world and subscribes to the philosophy of "*learning by being critiqued*" (Fischer et al., 1991). The task of the critic system is to present a reasoned opinion about a product or action sequence created by a learner in a self-directed manner (i.e. goal and action plan chosen by the learner). It uses a knowledge representation technique inherited from tutors (condition-action rules) but in a slightly relaxed form. Critics share with interface agents the characteristic of being able to independently execute processes and take action whenever they see an opportunity. Actions that are judged relevant to users are displayed in a critique window in the user interface.

More recently a "*second wave*" of educational agents has been proposed; these are characterized by their focus on interactive learning. We refer to them as pedagogical agents (Chen & Wassson, 2002; Jondahl & Mørch, 2002), a term originally proposed by Johnson et al. (1998), but slightly revised for our purposes. They define pedagogical agents as autonomous and/or interface agents that support human learning by interacting with students in the context of an interactive learning environment; however, in their work the context for the pedagogical agents is the ITS framework (Johnson, et al., 1998).

We see an important role for pedagogical agents as conceptual awareness mechanisms in CSCL environments. From our perspective there is convergence between the computational processes a pedagogical agent would be expected to do and the task and concept awareness mechanisms for educational groupware proposed by Gutwin et al. (1995). Furthermore, when the system-building efforts are grounded in a theory that has been influential in social psychology (Mead, 1934; Dodds &Valsiner, 1997) it provides a unique approach to pedagogical agents. In the next section we address the requirements for task and concept awareness in more detail and incorporate them as two dimensions of a four-dimensional, pedagogical agent design space.

3. A design space for pedagogical agents

We have found the following four dimensions to be of particular relevance for situating pedagogical agents for distributed collaborative learning environments: presentation, intervention, task, and pedagogy. We believe they can form a design space for others to follow; both for situating new developments and for stimulating conceptual improvements. In the next sub-sections these dimensions are described and illustrated by examples.

3.1 Presentation dimension

Presentation is an attribute of interface agents and not associated with software agents in general. Presentation is about how an agent should present itself to the users in the context of an interactive learning environment. There are many types of presentation techniques employed by agents to convey information to users, but perhaps the most well

known is the technique popularized by the Office Assistant (paper clip) agent in MS Office. This agent can guide a user toward creating a Word document or a PowerPoint presentation. Presentation techniques employed by this agent are graphics and simple animation. Other commonly employed presentation techniques are text, speech, and body language. Text includes messages presented in a permanent window on the screen or in a temporary pop-up window. Speech ranges from human voice recording to context-sensitive computer generated speech. Graphics can include still images or a sequence of images to convey motion. Research on animated graphical agents has spawned a great deal of interest in the software agent community and is now commonly regarded as a subfield of software agent research. Animated interface agents make use of techniques for simulating human body language, such as facial expressions, head nods, and gestures (Cassell, 2000). One of the first attempts at building a pedagogical agent system made use of animation techniques, such as body movements and gestures, to emphasize the areas of the interface students should pay attention to (Johnson, Rickel & Lester, 2000).

Although there is a large body of research in this area, there is no conclusive evidence that favors animated agents over non-animated agents when it comes to communicating information to human users for the purpose of increasing learning. Instead there are many interesting studies that give insightful information of the usefulness (or lack thereof) of pedagogical agents for specific learning situations. For example, Lester et al. (1997) found that animated agents led to the "*Persona effect*", which means the presence of a lifelike character in the user interface will have a positive effect on the learning experience. The study also found that combinations of types of advice could increase

learning performance. Other studies have found indications of the Persona effect, but no indications of increased learning performance (e.g. Craig, Gholson & Driscoll, 2002).

3.2 Intervention dimension

Intervention is about timing: when should the agent present itself to a user. Intervention is thus closely related to presentation and together they form two key characteristics of interface agents. The functionality of intervention can been seen as analogous to the operation of a thermostat. When a certain temperature value has been reached in a heat-controlled room, the thermostat reacts (e.g. turns the heat off). A programmed agent reacts similarly to events in its environment. When certain events are detected, such as a mouse click, a new user entering a session, modified workspace entries, or new items in a discussion forum, the agent responds by presenting a message or visual image in the interface. An example of an agent, developed according to this metaphor, was the "*design critic*" developed for the Janus design environment (Fischer, McCall & Mørch, 1989). These critics looked "*over the shoulder*" of a user and when they detected a "*critical state change*", they would immediately present their findings in a dedicated critic window, computed based on the design rules for how kitchen design units should be positioned relative to each other. Users' perception of these critics ranged from considering them very positive (in their role as reminders of design principles) to finding them annoying (they had a tendency to interrupt planned action). If the agents interfered too much, they would fail to provide sufficient help (Fischer et al., 1991).

Developers of agents need to decide upon the intervention strategy for their agents and the following issues should be taken into account:

- Degree of immediacy (how soon);

- Degree of repetition (how often);

- Degree of intrusiveness (block, superimpose, etc.); and

- Degree of eagerness (how important).

An intervention strategy must specify how soon to send information to users and how often it should be repeated in case of neglect. Intervention immediately after a trigger value has been detected (an immediate intervention strategy) has the advantage that the problem context is still active in the learners' mind. A disadvantage is that it may disrupt a potentially fruitful collaboration process that has just started. On the other hand, delayed messages may appear out of context and hence come too late for the feedback to have the desired effect. Messages that are repeatedly ignored may have to be placed at a location that will not disturb the user's main activity, or be postponed to a later time.

Agents can use various degrees of intrusiveness and eagerness to attract the users' attention. For example, an agent can force a user to attend to its information. When agents present information judged to be important, this may be the preferred strategy, but in other situations it will be perceived as annoying and distracting from the main activity. A less intrusive strategy is to display feedback as a separate process, i.e. in a separate window or by an animated character that is superimposed on the current process without disrupting it. The latter approach is the intervention strategy of the MS Office Assistant. This agent (animated character) runs parallel to the active process, but it is difficult to

ignore. The agent's graphic depiction reflects its degree of eagerness to provide assistance. An animated character that waves vigorously is easier to notice than one that maintains few states of displacement, but the former will most likely be turned off if it is not helpful.

A middle-ground intervention strategy is to provide the user with a choice whether to read and process the assistance immediately or first complete an action sequence in progress. In any case the messages by the agent should be displayed in such a way that they do not go unnoticed, and those messages pertaining to the user's current focus of attention should always be easy to find and not be hidden among a large set of information related to other parts of the environment.

3.3 Task dimension

A task in the context of a collaborative learning environment is often complex and multifaceted. It may specify individual work as well as joint work, which can be either well defined or ill defined. For example, tasks that involve the creation of domain-specific models will typically require different types of agent support than tasks that are open-ended and controversial, such as a learning task that asks students to "*discuss the pros and cons of genetically modified food.*"

It is notoriously difficult to create unambiguous tasks in complex social learning environments (Wasson, Guribye & Mørch, 2000). The number of factors to consider is often beyond the capacity of individual learners, and the computer support needs to be

supplemented with social protocol with the effect of local variability in task interpretation. However, this is not necessarily negative and many authors claim that agents combined with other types of scaffolds, such as sentence openers (Soller et al., 1999) and scientific discourse categories (Ludvigsen & Mørch, 2003), need to be assessed based on their potential to make tasks harder or simpler rather than trying to model the task completion process towards optimal performance.

Previous studies have shown that scaffolds can change complex tasks in various ways, most notably by simplifying them (Norman, 1987). For example, the MS Office Assistant can guide the user through the task of formatting a document (such as indenting an enumerated list in non-standard position) by providing the right information at the right time. However, recent studies have shown that making the task *harder* can also have positive effects, especially on learning. Reiser (2002) has shown in a series of studies that learning environments augmented with scaffolds designed to make concepts more problematic (requiring students to question their contributions) and restructure the task can improve learning by providing alternative opportunities for productive problem solving. Similarly, Fischer (1994) argues that critic systems that create "*breakdown*" in the users' interaction with a learning environment can stimulate creativity in performing the task (by temporarily suppressing an action sequence and forcing an evaluation, or requiring a work-around in ways that were previously unthinkable). The rationale behind this is not to hinder activity but to shorten the action-evaluation loop and support reflection-in-action (Fischer, 1994).

Designers of pedagogical agents need to pay attention to the type of tasks a learning environment is intended to support, both individual and shared aspects and the degree of domain-specificity involved, and then decide whether pedagogical agents should simplify or constructively hinder (make more problematic) the completion of these tasks.

3.4 Pedagogy dimension

Pedagogy in this context is about the kind of material to be presented to learners by an agent in a distributed learning environment. Identifying and supporting this dimension of an agent can be challenging. We address it by giving agents the role of monitoring and organizing ongoing collaborative processes such as participation, group interaction, coordination, teacher intervention, and scientific discourse in situations where it is difficult for human facilitators to get an overview because the learners are geographically distributed and/or work at different hours. It can be approximated in two ways:

- Making learners aware of their own and other learners' actions and activities; and
- Identifying the '*common attitude*' (Mead, 1934) and shared principles of a group of learners.

These two strategies are related, since the latter will often presuppose the former. By monitoring the shared information space of an online learning environment an agent can automatically compute statistics based on who is logged on, who communicates with whom, what objects they act upon, how much of a shared task has been completed, etc. This can then be presented to instructors as well as students. However, a pedagogical agent will often go a step further and present the information as a "*generalized piece of advice*" associated with the computed information. This information is more than the sum

of individual learners' actions and activities in the learning environment and will therefore not be determined solely on the basis of the collected data. It needs to be integrated with other sources of information, such as paradigmatic examples, models and principles associated with teaching styles, cooperative interaction, and scientific discourse. We address this form of awareness from the points of view of collaboration principles and knowledge building.

3.4.1 Collaboration principles

In systematic (well-defined) domains there are shared principles and rules of behavior acknowledged by experts as useful for others to know in order to successfully navigate within the space of potential moves. Although these principles and rules are harder to identify in less stringently structured domains such as collaborative learning, researchers in CSCW have identified recurring relationships (role and rule combinations) that need to be satisfied before a shared task among a group of people can successfully be completed. These include common goals, shared feedback (Dourish & Bellotti, 1992), mutual learning (Bratteteig, 1997), and genuine interdependence (Salomon, 1992).

In a study conducted by Salomon (1992), it was found that the quality of students' collaboration correlated with their degree of mindfulness during the process. Mindful engagement in the task, he claims, plays a crucial role if one wants to achieve successful group interaction. To achieve this, the environment has to be organized to allow opportunities for '*genuine interdependence*' to exist, which means the group members must make an effort to:

- Share information;

- Divide labor; and

- Think jointly in explicit terms.

Supporting genuine interdependence is not something an agent can do on its own. It needs to be coordinated by instructor participation. However, an agent can make the students aware of the other members' presence in the environment as well as their activities, and it can suggest that a group of people who work on the same task should divide labor if they work on the same shared object for a prolonged time without any improvement.

3.4.2 Knowledge building

Knowledge Building (Scardemalia & Bereiter, 1994) and its subsequent refinement, Progressive Inquiry (Hakkarainen, Lipponen, & Järvelä 2002) are influential pedagogical models in CSCL. Knowledge building requires that new knowledge is not simply assimilated with the help of a more knowledgeable person, but also jointly constructed through solving problems with peers by a process of building shared understanding. One reason for the popularity of these models is that they fit well with the educational philosophy instituted by many schools in Canada and Scandinavia (problem-based learning), as well as elsewhere in the world. The basic idea is that students gain a deeper understanding of a knowledge domain by engaging in a research-like process in the domain by generating their own problems or questions, proposing tentative answers (working theories) and searching for deepening knowledge collaboratively.

The regularity of knowledge building is manifest in a set of scientific discourse categories to be chosen each time a message is posted by a student in the online discussion forum of systems such as Future Learning Environment (FLE) (Muukkonen, Hakkarainen & Lakkala, 1999). Scaffolding knowledge building includes: 1) helping students to choose what category to use when proposing a '*problem*', 2) teaching students what a '*working theory*' is, 3) helping them use a working theory in relation to a problem or a '*scientific explanation*', 4) finding information to back up working theories, and 5) collaborating with others during these processes (Ludvigsen & Mørch, 2003).

Developers of pedagogical agents use the pedagogy dimension to identify the information to be contained in the advice presented by the agents to the users of a virtual learning environment. Principles, examples, models, rules, and roles of collaboration and knowledge building are sources of this information.

In the next section we present three systems aimed at scaffolding collaboration and knowledge building according to the above criteria. We present one system in detail and summarize the other two.

4. Collaboration Patterns Agent Simulation

The first pedagogical agent we created was Collaboration Patterns Agent Simulation (CoPAS) (Jondahl & Mørch, 2002), a simulation study carried out with the Wizard-of-Oz technique (e.g. Dahlbäck, Jönsson & Ahrenberg, 1993). The goal was to simulate agent functionality in a groupware system without first implementing it. Human experts acted

as agents (wizards) and gave assistance to users regarding tool use, task understanding and remote collaboration. The empirical results have enabled us to build software functionality in the subsequent efforts.

4.1. TeamWave Workplace

The learning environment employed, TeamWave Workplace (TW) (Roseman & Greenberg, 1996), is a synchronous groupware system providing users with a set of shared tools for working and learning together. TW is based on the room metaphor and the tools can "*furnish*" virtual places, or rooms inside the environment. The rooms can be configured independently and according to one's needs. The tools most frequently used by our students were Concept Map, Whiteboard, Chat, Post-it notes, Message Board, To-Do list, and Calendar. Figure 2 shows a Workplace room with a configuration of some of those tools. In particular, the Concept map, Chat and Post-it notes were used as part of the design exercise we describe below.

Insert Figure 2 about here.

4.2 Method

Six groups of freshmen students from the Information Science Department at the University of Bergen participated in the study. They were organized into groups of threes and each group was given the task of creating object-oriented analysis and design (OOA/D) diagrams for an Internet banking system. The session lasted for 90 minutes. The fixed time duration was judged appropriate because the assignment was taken from a previous course exam. Furthermore, the wizards could possibly reveal themselves if they

were active for a longer period of time, since the students would be able to guess they were part of an agent simulation study.

We used the Wizard of Oz technique to simulate agent functionality. The participants were led to believe they were interacting with a software agent, when in fact they were interacting with a simulation staged by human experts (Dahlbäck, et al., 1993; Maulsby, Greenberg & Mander, 1993). The Wizard of Oz technique was chosen because it is a frequently used method for testing new ideas without first implementing them in software. When using state of the art collaboration technology, the cost of implementing new system functionality can be high, especially when existing systems are not built for radical customization. With the Wizard of Oz technique we have been able to make a realistic simulation of a collaborative learning environment in a shorter time than it would have taken us if we were to program the agents in a programming language. The method has allowed us to try out functionality that we later refined in software prototypes (Section 5).

Three graduate students acted as wizards, simulating the following agent functionality: 1) Tool Agent, 2) Domain Agent and 3) Collaboration Agent. The tool agent had technical knowledge about how to use the TW tools, the domain agent provided assistance on object-oriented analysis and design, and the collaboration agent provided assistance on how to collaborate based on pedagogical models (Koschmann, 1996), awareness (Gutwin, et al., 1995), and collaboration patterns (Salomon, 1992; Wasson & Mørch, 2000). The wizards were given oral instructions for how and when to act and had a list of

rules for what to say upon acting (tool usage, domain and collaboration principles). The intervention technique was to react immediately, but the wizards had to perform a set of manual operations that could delay the feedback up to a few minutes. The presentation technique employed were text messages displayed in pop-up boxes, a technology readily available in TW as a command in its designer substrate. An example of an agent's message is shown in Figure 3.

Insert Figure 3 about here.

4.3 Findings

Ericsson (1999) has identified three factors that he judged as important when measuring the quality of feedback provided by the wizards in a Wizard of Oz simulation study:

- Type of feedback;
- Number of instances of each type; and
- Time delay in deliverance.

The type of feedback in this case refers to the range of message alternatives that the user can choose from.

The CoPAS trial was conducted in such a way that we could measure the first two factors, but not the last. It was not possible to monitor the time delay between user request and agent response, but we have data on the other two factors (Table 1).

Insert Table 1 about here.

Table 1 shows the total number of messages sent by the wizards to the six groups. Of the original 38 message alternatives they could choose from, 25 (65.5%) were used at least once. In Ericsson's (1999) study he reported that 54% (18 of 33) of the messages were used. The overall picture is that there is a relatively good match between the type of messages prepared in advance and the messages actually used.

One the other hand, the wizards created a total of 19 new messages during the course of the trial. They were not told to do so, nor were they prevented from doing it. During the interviews afterward they told us they had to make up these messages when the students were stuck in difficult situations for which there were no predefined messages to assist them.

4.4 Analysis

The data below is an excerpt[1] from a chat communication between the three members of one of the groups. Student 1.1 sends a question to the other two in her group regarding the definition of (object-oriented) classes. She attempts to answer it herself, but later receives an answer from Student 1.3 (Line 3-4). Then, an agent intervenes (Line 5) and the dialog shifts to another level of abstraction:

    1. Student 1.1: Where in the diagram do we write in class attributes?

---

[1] This and subsequent excerpts have been translated from Norwegian by one of the authors. The numbered lines across the two excerpts are for the current analysis only. They do not follow each other in the data set.

2. Student 1.1: ..in the yellow frame[2]?

3. Student 1.3: I have already done it in the yellow rectangles on the top

4. Student 1.3: I was able to change the name in both of the classes

5. *Domain agent: Is it possible to make a super-class here?*

6. Student 1.3: Should we add another class?

7. Student 1.3: Yeah.., let's make a common super-class!

8. Student 1.2: How?

9. Student 1.3: We can call it Customer!

In Line 5 the Domain-agent intervenes and provides a suggestion for a new class to be included in the diagram based on the classes they have already created. As a result they shift their focus from a low-level operation (class-definition mechanics) to a higher-level, domain-oriented concept ("*Customer*" as a super-class name). This shift can be seen as a step towards generalization, which is a desired behavior in many learning environments, i.e. being able to create higher-level abstractions from lower level details. Although it is premature for us to make conclusions based on the above example, it points out a role for pedagogical agents. They can operate in the "*middle ground*", between concrete examples and the symbols of a task description when there are well-defined connections between the two levels. Furthermore, an agent can alternatively cause a "*breakdown*" or simplify the flow of interaction. This can effect the completion of the assigned task, as well as the learning outcome in some cases. In the situation above the students were able to move faster toward a solution of the assignment given to them.

---

[2] What Student 1.1 calls "yellow frame" and Student 1.3 calls "yellow rectangle" are TW Post-it notes. They were used for writing object-oriented class definitions.

The feedback from the agents was generally well received by the students, but in one particular situation this was not the case. One of the messages frequently issued by the Collaboration agent was about '*division of labor*' (see Figure 3). Receiving this message had a negative impact on the collaboration in one of the groups, as illustrated by the following excerpt:

10. *Collaboration agent: It can be useful to divide the work*

11. Student 3.1: we should divide the tasks

12. Student 3.1: I can start with the class diagram...

13. Student 3.3: what about me, what should I do?

There were two tasks to be completed in the assignment (class diagram and use case diagram). Since each group had three members there was no equal way of dividing the tasks. The comment on line 13 was discussed with the students in the final interview. It turned out two of them knew each other quite well, but the third did not know the other two (not an uncommon situation in remote collaboration). As a consequence she felt left out of much of the activity. Furthermore, one of the students of that group gave the following answer when he was asked if knowing each other influenced the division of work in their group: "*It certainly influenced our work because you know how to talk to someone you already know, and that can complicate collaboration with a third person who is supposed to be part of solving the same assignment.*" Whether or not the agent contributed to the alienation of the third member in this group can only be speculated, but

it does indicate that agent feedback may have both positive and negative effects on group interaction. A human facilitator (teacher, instructor) would resolve the above situation in a more appropriate manner.

Based on the findings from the CoPAS study (Jondahl & Mørch, 2002), we have identified the following issues for further work (the list is not exhaustive, but identifies the problems we chose to work on):

- Real-time (synchronous) groupware should be augmented by asynchronous collaboration support;

- Well-defined learning tasks should be supplemented by open-ended tasks;

- Pedagogical agents should be customizable by end users; and

- Instructors should be included in the loop to resolve ambiguities in tool usage, task understanding, and remote collaboration.

We address the first three issues in the next section (subsequent work) and the fourth issue in the following section (related work).

5. Subsequent work

To follow up on the lessons learned from the CoPAS study, we have built an agent for an asynchronous collaborative learning environment, Student Assistant Agent (SA-Agent), and next made it customizable, so the agent rules can be modified by end users (e.g. instructors).

5.1 Student Assistant Agent

The Student Assistant (SA) Agent (Dolonen, Chen & Mørch, 2003) is a pedagogical agent for the FLE learning environment (Leinonen, 2003). The collaborative knowledge-building process with FLE is dependent on the use of message categories when posting notes in the knowledge-building forum. The rationale behind this was described in a previous section (Section 3.3.2). However, it can be difficult for a student to understand how to use these categories (Ludvigsen & Mørch, 2003). In order to lessen these problems, we designed and developed an agent system (Mørch, Dolonen & Omdahl, 2003) that can monitor the interaction process, analyze the information collected and provide the students with advice on the collaboration and knowledge-building activity. When monitoring the process, the agent system gathers statistical information of the user activity and stores it in a database. This information includes the structural properties of the messages posted by the students. Based on this information, the agent can provide the following kind of advice: 1) encourage in-active students to be more active, 2) suggest what messages to reply to and who should be doing so, 3) suggest what category to choose for the next posting, 4) suggest when messages do not follow the recommended knowledge building steps .

5.2 Rule Editor

Previous studies indicate agents and other type of computational scaffolds can be brittle and not well adapted to the local needs of users and their unique contexts (e.g. Ludvigsen & Mørch, 2003). The effect of failing to provide assistance can be that agent messages are ignored or inhibit productive interaction. This dilemma is especially pertinent in

poorly structured knowledge domains because the chances for an agent to '*miss the point*' are much greater than in more well-defined domains. For example, during the CoPAS session the agents (wizards) made up new rules when the students were stuck in difficult situations. In total 19 out 59 rules (33%) had to be created on the fly during the session. These findings indicate agent rules should be editable at run time. At this point the interaction with the agent is still fresh in the mind. Similarly, one of the limitations of the SA-Agent is brittleness of its rules. This is a consequence of predefined trigger values (hard-coded into the system) and feedback messages with fixed (non-context sensitive) message bodies.

The Rule-Editor (Nævdal, 2003) addresses some of these shortcomings. It interacts with the SA-Agent's database in order to access its rules and variables. The Rule-Editor is activated in the FLE's user interface through the agent's presentation mechanism. When a user chooses to edit a rule, she signals this by invoking an "*Edit Rule*" command (with a single mouse click). A wizard guides the user through the task of modifying the various aspects of a rule (e.g. trigger values, rule firing priorities, display message). This results in new display values and intervention intervals for the agent and hence provides a remedy for its perceived brittleness (Nævdal, 2003). The '*language*' for tailoring is remarkably simple because it is well defined and task-specific from the point of view of end-user development. We believe it is well suited for advanced users, such as instructors and students who are familiar with using computers. However, this is currently a conjecture and will be explored in future work.

Table 2 summarizes our efforts in developing pedagogical agents for distributed collaborative learning environments. The table presents agents in the columns and agent dimensions in the rows.

Insert Table 2 about here.

6. Related Work

GRACILE (Japanese GRAmmar Collaborative Intelligent Learning Environment) (Ayala & Yano, 1996) supports the teaching of Japanese language to foreign students in Japan. The system has an intelligent agent component that assists the students with a collaborative-learning task in a virtual community of practice. A mediator-agent assists the students with tasks that require them to make use of their collaboration potential in their interactions with each other. It is founded on the theory of proximal development originally proposed by Vygotsky (1978); our work shares their educational philosophy.

Soar Training Expert for Virtual Environments (STEVE) and Agent for Distance Learning: Light Edition (ADELE) are animated pedagogical agents (Johnson et al., 2000). STEVE is integrated with a virtual reality environment for modeling US naval ships and how to operate them. ADELE operates in a web-based learning environment promoting courses in medical and dental education. It can provide feedback to students by offering hints or rationale for taking particular actions. Both agents use speech and body movement as presentation techniques and can monitor individual students' progress

towards well-defined goals within the domains. As such they have more in common with ITS than CSCL. Our use of the term pedagogical agents has been inspired by this work.

Encouraging Positive Social Interaction while Learning ON-Line (EPSILON) (Soller, et al, 1999) is a project that has investigated the integration of intelligent facilitation agents with a shared workspace of object-oriented analyses and design (OOA/D). These agents can observe a group's conversation and dynamically analyse individual student contributions. The dialog among students is scaffolded by sentence-openers modelled on speech act theory (e.g. justify, assert, encourage). The EPSILON agents are able to recognize events, such as a student completing a critical portion of the task, or a student failing to discuss his or her actions with others. When it detects an opportunity to react, the agent may intervene by asking the group to explain the student's actions. Both CoPAS and SA-Agent adopt strategies from EPSILON: CoPAS makes use of a similar taks domain (OOA/D) and the SA-Agent makes use of computer-supported scaffolding resembling sentence-openeners (FLE knowledge-building categories).

Fischer and Girgensohn (1990) have created an end-user modification component for the Janus system. MODIFIER enables an end user to modify the critiquing rules with a high-level tailoring language. A new rule is defined with the "*New Rule*" command. A property sheet gives the user a template for selecting values that need to be in place for a new rule to become operational within the environment. The system gives additional support for the specification of rule conditions. A rule condition consists of relationships between design units. These can be selected from a menu of choices and, if none of the

existing ones apply, a new one can be created. We have created a similar tailoring language for the wizards in the Rule-Editor agent.

The Instructional Assistant (IA) Agent (Chen & Wasson, 2002; Dolonen et al., 2003) is another agent integrated with FLE. The IA-agent has two roles: 1) observe the distributed collaborative learning process and compute statistical information for viewing, and 2) detect possible problems in the interaction and present them to the instructor so that the instructor, if desired, can give feedback to the students. It employs the same system architecture as the SA-Agent. The reason for including the instructor in the loop is to avoid the situation wherein the agent's understanding of the collaboration process precedes human judgment. This may lead to misinterpretation or misunderstanding among the students. Instructors can review the the information before it is sent to the students and thus circumvent this problem.

7. Summary and Conclusions

We have described a series of efforts to integrate pedagogical agents with collaborative environments (synchronous and asynchronous). We employed theory-based systems engineering and a trial session and evaluation to explore how pedagogical agents can be useful in collaborative learning environments. Our system building efforts have reused and extended previous efforts, both our own and others. Observations have provided us with feedback to direct subsequent work in this area. The agents function as an extended awareness mechanism, supporting conceptual awareness, which is motivated by Mead's theory of the '*Generalized Other*'.

Our contributions are summarized as follows: 1) a design space for classifying pedagogical agents: presentation, intervention, task, and pedagogy, 2) a series of attempts that shows it is possible to take advantage of statistical information in collaborative learning environments without detailed student modeling, 3) an approach to represent '*common attitude*' and principles associated with collaborative performance, and 4) customizable agents to address the imprecision dilemma associated with providing agent-based assistance in poorly structured knowledge domains.

Acknowledgements

References

Anderson J, Boyle C, Farrell, R, Reiser B. Cognitive Principles in the Design of Computer Tutors. In: Morris P, ed. *Modeling Cognition*. New York, USA: John Wiley, 1987.

Apple Computer. Knowledge Navigator Video. 1997. Available at (2003-09-13)

http://www.billzarchy.com/clips/clips_apple_nav.htm

Ayala G, Yano Y. Intelligent Agents to Support the Effective Collaboration in a CSCL

Environment. In: *Proceedings of World Conference on Educational*

Telecommunication (ED-TELECOM'96), June 17-22, Boston, MA, USA: AACE

Press, 1996:19-24.

Baggetun R, Dolonen J, Dragsnes, S. Designing Pedagogical Agents for Collaborative

Telelearning Scenarios. In: Bjørnestad S, Moe R, Mørch A, Opdahl, ed.

Proceedings of the 24th Information Systems Research Seminar in Scandinavia

(IRIS'24), Aug. 11-14, Ulvik, Norway: Department of Information Science,

University of Bergen, 2001:29-42.

Bannon L. Computer-Mediated Communication. In: Norman DA, Draper SW, ed. *User*

Centered System Design: New Perspectives on Human-Computer Interaction.

Hillsdale, NJ, USA: Lawrence Erlbaum, 1986:433-456.

Bates J. Virtual Reality, Art, and Entertainment. *Presence: Teleoperators and Virtual*

Environments 1992;1(1):133-138.

Beaudouin-Lafon M, Karsenty A. Transparency and Awareness in a Real-Time

Groupware System. In: Green M, Mackinlay J, ed. *Proceedings of the ACM*

Symposium on User Interface Software and Technology (UIST'92), Nov. 15-18,

Monterey, CA, USA: ACM press, 1992:171-180.

Bratteteig T. Mutual Learning: Enabling Cooperation in Systems Design. In: Braa K,

Monteiro E, ed. *Proceedings of the 20th Information Systems Research Seminar*

in Scandinavia (IRIS'20), Aug. 9-12, Hankø, Norway: Department of Informatics, University of Oslo, 1997:1-20.

Burton RR, Brown JS. An Investigation of Computer Coaching for Informal Learning Activities. In: Sleeman D, and Brown, JS, ed. *Intelligent Tutoring Systems*. New York, USA: Academic Press, 1982:79-98.

Cassell J. Embodied Conversational Interface Agents. *Communications of the ACM* 2000;43(4):70-78.

Chen W, Wasson B. An Instructional Assistant Agent for Distributed Collaborative Learning. In: Cerri, S, Gouarderes G, Paraguacu, F, ed. *Proceedings of Intelligent Tutoring Systems Conference (ITS 2002)*, June 2-7, Biarritz, France and San Sebastian, Spain: Springer Lecture Notes in Computer Science, 2002: 609-618.

Craig SD, Gholson B, Driscoll D. Animated Pedagogical Agents in Multimedia Educational Environments: Effects of Agent Properties, Picture Features, and Redundancy. *Journal of Educational Psychology* 2002;94:428-434.

Dillenbourg P, Baker M, Blaye A, O'Malley C. The Evolution of Research on Collaborative Learning. In: Spada E, Reiman P, ed. *Learning in Humans and Machine: Towards an Interdisciplinary Learning Science.* Oxford, England, UK: Elsevier, 1996:189-211.

Dahlbäck N, Jönsson A, Ahrenberg L. Wizard of Oz Studies - Why and How. In: Gray WD, Hefley WE, Murray D, ed. *Proceedings of the 1993 International Workshop on Intelligent User Interfaces (IUI 1993),* Jan. 4-7, 1993, Orlando, FL, USA: ACM Press, 1993:193-200.

DeSanctis G, Gallupe RB. A Foundation for the Study of Group Decision Support Systems. *Management Science* 1987;33(5):589-609.

Dodds AE, Valsiner J. The Personal and the Social: Mead's Theory of the 'Generalized Other'. *Theory & Psychology* 1997;7(4):483-503.

Dolonen J, Chen W, Mørch A. Integrating Software Agents with FLE3. In: Wasson B, Ludvigsen S, Hoppe U, ed. *Proceedings of the International Conference on Computer Support for Collaborative Learning 2003* (*CSCL 2003)*, June 14-18, Bergen, Norway: Kluwer Academic, 2003:157-161.

Dourish P, Bellotti, V. Awareness and Coordination in Shared Workspaces. In: Greenberg S, ed. *Proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW'92)*, Oct. 31- Nov. 4, Toronto, Canada: ACM Press, 1992:107-114.

Ellis CA, Gibbs SJ, Rein GL. Groupware: Some Issues and Experiences, *Communications of the ACM* 1991;34(1):39-58.

Ericsson M. *Supporting the Use of Design Knowledge.* Dissertation No. 592, Department of Computer and Information Science, University of Linköping, Sweden, 1999.

Fischer G, Lemke AC, Mastaglio, T, Mørch AI. The Role of Critiquing in Cooperative Problem Solving. *ACM Transactions on Information Systems* 1991;9(3):123-151.

Fischer G. Turning Breakdowns into Opportunities for Creativity. *Knowledge-Based Systems* 1994;7(4):221-232.

Fischer G, Girgensohn A. End-User Modifiability in Design Environments. In: Lynch G, Meads J, ed. *Proceedings Human Factors in Computing Systems (CHI'90)*, April, Seattle, WA, USA: ACM Press, 1990:183-191.

Fischer G, McCall R, Mørch A. Janus: Integrating Hypertext with a Knowledge-based Design Environment. In: Akscyn R, Halasz F, ed. *Hypertext 89 Proceedings*, Nov. 5-8, Pittsburgh, PA, USA: ACM Press, 1989:105-117.

Grudin J. Computer-Supported Cooperative Work: History and Focus. *IEEE Computer* 1994;27(5):19-26.

Gutwin, C., Stark, G., Greenberg, S. (1995). Support for Workspace Awareness in Educational Groupware. In: *Proceedings of the ACM Conference on Computer Supported Collaborative Learning (CSCL'95)*, Oct. 17-20, Bloomington, Indiana, USA: ACM Press, 1995:147-156.

Hakkarainen K, Lipponen L, Järvelä S. Epistemology of Inquiry and Computer-Supported Collaborative Learning. In: Koschmann T, Hall H, Miyake N, ed. *Computer Supported Collaborative Learning 2: Carrying Forward the Conversation*. Mahwah, NJ, USA: Lawrence Erlbaum, 2002:129-156.

Hiltz SR, Turoff M. Structuring Computer-Mediated Communication Systems to Avoid Information Overload. *Communications of the ACM* 1985;28(7):680-689.

Huber G. Organizational Information Systems: Determinants of their Performance and Behavior. *Management Science* 1982;28(2):138-153.

Hughes JA, Randall D, Shapiro D. Faltering from ethnography to design. In: Greenberg S, ed. *Proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW'92)*, Oct. 31- Nov. 4, Toronto, Canada: ACM Press, 1992:115-122.

Jondahl S, Mørch A. Simulating Pedagogical Agents in a Virtual Learning Environment. In: Stahl, G, ed. *Proceedings Computer Support for Collaborative Learning (CSCL 2002)*, Jan. 7-11, Boulder, CO, USA: Lawrence Erlbaum, 2002:531-532.

Johnson WL, Rickel JW, Lester JC. Animated Pedagogical Agents: Face-to-Face Interaction in Interactive learning Environments. *International Journal of AI in Education* 2000;11:47-78.

Johnson WL, Rickel J, Stiles R, Munro A. Integrating Pedagogical Agents into Virtual Environments. *Presence: Teleoperators and Virtual Environments* 1998;7(6):523-546.

Johnson-Lenz P, Johnson-Lenz T. Groupware: The Process and Impacts of Design Choices. In: Kerr EB, Hiltz SR, ed. *Computer Mediated Communication Systems: Status and Evaluation.* New York, USA: Academic Press, 1982:44-55.

Kay A. Computer Software. *Scientific American* 1984;251(3):41-47.

Kiesler S, Siegel J, McGuire T. Social Psychological Aspects of Computer Mediated Communication. *American Psychologist* 1984;39:1123-1134.

Koschmann T. Paradigm Shifts and Instructional Technology: An Introduction. In: Koschmann T, ed. *Computer Supported Collaborative Learning: Theory and Practice of an Emerging Paradigm.* Mahwah, NJ, USA: Lawrence Erlbaum, 1996:1-23.

Leinonen T. Fle3: Future Learning Environment. Website hosted by Media Lab, University of Art and Design Helsinki. Available at (2003-08-20) http://fle3.uiah.fi/

Lester J, Converse S, Kahler S, Barlow T, Stone B, Bhogal R. The Persona Effect: Affective Impact of Animated Pedagogical Agents. In: Pemberton S, ed. *Proceedings of Human Factors in Computing Systems (CHI* '97), March 22-27, Atlanta, GA, USA: ACM Press, 1997:359-366.

Lieberman H, Selker T. Agents for the User Interface. In: Bradshaw J, ed. *Handbook of Agent Technology*. AAAI/MIT Press, 2003 (to appear).

Liu J, Zhong N, Yao Y, Ras ZW. The Wisdom Web: New Challenges for Web Intelligence (WI). *Journal of Intelligent Information Systems* 2003;20(1):5-9.

Ludvigsen S, Mørch A. Categorisation in Knowledge Building: Task-specific Argumentation in a Co-located CSCL Environment. In: Wasson B, Ludvigsen S, Hoppe U, ed. *Proceedings of the International Conference on Computer Support for Collaborative Learning 2003* (*CSCL 2003)*, June 14-18, Bergen, Norway: Kluwer Academic, 2003:67-76.

Maes P. Agents that Reduce Work and Information Overload. *Communications of the ACM* 1994;37(7):31-40.

Maulsby D, Greenberg S, Mander R. Prototyping an Intelligent Agent Through Wizard of Oz, In: *Proceedings of Conference on Human Factors in Computing Systems 1993 (INTERACT'93 and CHI'93)*, April 26-29, Amsterdam, The Netherlands: ACM Press, 1993:277-285.

McCarthy J. Programs with Common Sense. *Technical report*. Department of Computer Science, Stanford University, 1959. Available at (2003-09-13) http://www-formal.stanford.edu/jmc/mcc59/mcc59.html

Mead GH. *Mind, Self, and Society*. Chicago, IL, USA: University of Chicago Press, 1934.

Muukkonen H, Hakkarainen K, Lakkala M. Collaborative Technology for Facilitating Progressive Inquiry: Future Learning Environment Tools. In: Hoadley C, Roschelle J, ed. *Proceedings of the Computer Support for Collaborative Learning (CSCL) 1999 conference.* Dec. 12-15, Palo Alto, CA, USA: Lawrence Erlbaum, 1999:406-415.

Mørch A, Dolonen J, Omdahl K. Integrating Agents with an Open Source Learning Environment. In: Lee KT, Mitchell K, ed. *Proceedings of International Conference on Computers in Education 2003 (ICCE 2003),* Dec. 2-5, Hong Kong: AACE Press, 2003 (to appear).

Norman DA. *The Psychology of Everyday Things.* New York, USA: Basic Books, 1987.

Nwana HS. Software Agents: An Overview. *The Knowledge Engineering Review* 1996;11(3):1-40.

Nævdal JE. User Tailorable Pedagogical Agents for a Groupware System. Master's thesis. Department of Informatics, University of Oslo, 2004 (to appear).

Reiser B. Why Scaffolding Should Sometimes Make Tasks More Difficult for Learners. In: Stahl, G, ed. *Proceedings Computer Support for Collaborative Learning (CSCL 2002)*, Jan. 7-11, Boulder, CO, USA: Lawrence Erlbaum, 2002:255- 264.

Roseman M, Greenberg S. TeamRooms: Network Places for Collaboration. In: *Proceedings of the 1996 ACM Conference on Computer Supported Cooperative Work (CSCW'96)*, Nov. 16-20, Boston, MA, USA: ACM Press, 1996:325-333.

Salomon G. What Does the Design of Effective CSCL Require and How Do We Study its Effects? *SIGCUE Outlook* 1992;21(3):62-68.

Scardamalia M, Bereiter C. Computer Support for Knowledge-Building Communities. *The Journal of the Learning Sciences* 1994:3(3):265-283.

Scully J. The Relationship Between Business and Higher Education: A Perspective on the 21st Century. *Communications of the ACM* 1989;32(9):1056-1061.

Selker T. COACH: A Teaching Agent that Learns. *Communications of the ACM* 1994;37(7):92-99.

Soller A, Linton F, Goodman B, Lesgold A. Toward Intelligent Analysis and Support of Collaborative Learning Interaction. *Proceedings of the Ninth International Conference on Artificial Intelligence in Education*, June, LeMans, France: IOS Press, 1999:75-82.

Vygotsky LS. *Mind in Society.* Cambridge, MA, USA: Harvard University Press, 1978.

Wasson B, Mørch A. Identifying Collaboration Patterns in Collaborative Telelearning Scenarios. *Educational Technology & Society* 2000;3(3). Available at (2003-10-31) http://ifets.ieee.org/periodical/vol_3_2000/c04.html

Wasson B, Guribye F, Mørch A. Project DoCTA: Design and use of Collaborative Telelearning Artefacts. *ITU Research Report 5*. Oslo, Norway: Unipub, 2000.

CoPAS wizards	Message alternatives	Message actually used	Number of messages sent to all five groups
Tool Agent	23 *(12 new)*	82.5%	42
Domain Agent	19 *(6 new)*	63%	34
Collaboration Agent	15 *(1 new)*	86.5%	56
All wizards	57 *(19 new)*	77%	132

Table 1. Agent message types and number of times they fired. The *new* items refer to the

number of messages created during the course of the session.

Dimension/ SYSTEM	Presentation	Intervention	Task	Pedagogy
CoPAS	Text in a pop-up box	Approximately immediate, interruptive	Object-oriented analysis/design	Division of labour, shared feedback
SA-AGENT	Text in a permanent window	Immediate, separate process	Knowledge building, PI category usage	Participation, knowledge building
RULE-EDITOR	Animated icon with text in temporary window	Immediate, separate process	Customize agent rules with wizard	Instructor participation

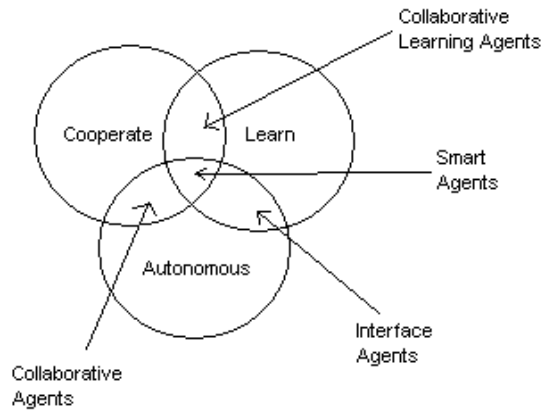**Table 2.** Summary of pedagogical agent characteristics

Figure 1. A typology of agents, adopted from Nwana (1996)
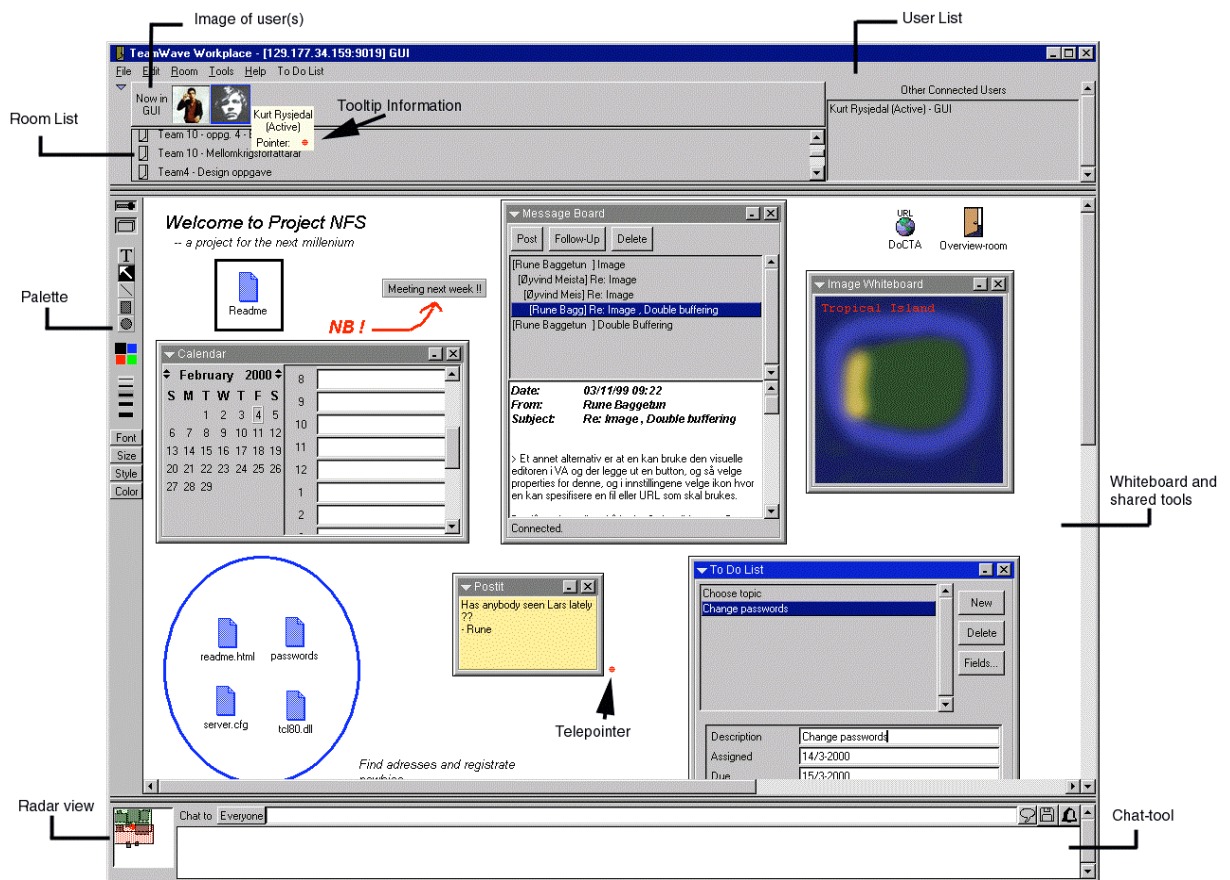

Figure 2. TeamWave Workplace

Figure 3. A pop-up box from the Collaboration agent shows a message regarding division of labor. The message is written in Norwegian and reads: "It can be useful to divide the work".