Supporting Discourse in a Synchronous Learning Environment: The Learning Protocol Approach

Hans-Rüdiger Pfister
Knowledge Media Research Center (KMRC), Tübingen, Germany
r.pfister@iwm-kmrc.de

Martin Mühlpfordt
Fraunhofer IPSI, Darmstadt, Germany
mamue@ipsi.fhg.de

ABSTRACT
Synchronous discussions in distributed computer-supported learning environments are usually conducted employing chat tools. Lack of coordination and coherence among the contributions is a typical problem with chat. In this paper, we propose to apply learning protocols to increase coordination, coherence, and as a consequence the efficiency of learning. Learning protocols are system controlled cooperation scripts, which impose constraints on the learning discourse: Participants explicitly identify the reference of their contributions as well as the type of contribution; furthermore, the order of contributions is predetermined. Learning protocols can be defined with respect to a variety of typical discourse situations, such as giving an explanation, or summarizing text. We describe two learning protocols and present empirical results which confirm the hypothesis that structured discourse leads to superior learning compared to unstructured chat.

Keywords
Cooperative learning, learning protocols, cooperation scripts, structured discourse

INTRODUCTION
Net-based cooperative learning is becoming more and more popular. Synchronous as well as asynchronous cooperative learning scenarios are easily realized using internet technology. Also, users, i.e., learners, teachers, moderators, become increasingly skilled utilizing these technologies. The most common technique for asynchronous communication is email, and for synchronous communication some kind of text-based chat is usually applied. In this paper, we will focus on the usefulness of synchronous text-based chat for cooperative learning discourses.

Scripted Cooperation
In traditional face-to-face classroom settings, the effectiveness of cooperative learning is a well established fact (Slavin, 1995). Cooperation can be further improved using so-called scripted cooperation. Scripted cooperation implies the application of a more or less rigid schema (a cooperation script), i.e., a set of rules and phases according to which the cooperation proceeds (O’Donnell & Dansereau, 1992). Imagine as a simple example a discussion group, moderated by a teacher who tries to structure the discussion along a sequence of specific phases, e.g., brainstorming, critique, and summary. A more sophisticated cooperation script has been proposed and tested by O’Donnell and Dansereau (1992). The objective of the cooperation is to summarize a text, consisting of several sections. For each section, one member of the learning group is assigned the role of a “summarizer”, whose task is to produce a concise summary of the current section. The other members of the group take the role of “commentators”, commenting and criticising the summary. With the following sections of the text, roles are switched and another learner is assigned the role of a summarizer; this continues until the whole text is processed. Other types of scripted cooperation have been studied by Slavin (1995), Mancini et al. (1998), or Huber (1999). Empirical evidence with respect to positive effects on knowledge acquisition and cooperation competence is mixed, though generally supportive (Huber, 1999; O’Donnell et al., 1987; Slavin, 1995).
Net-based Cooperation

In net-based cooperative learning situations, cooperation scripts might be even more useful. In virtual discussions, especially when using synchronous text-based chat-tools, several problems are prevalent. According to Hesse, Garsoffky and Hron (1997), some of the most important problems are: (i) lack of social awareness, i.e., due to the restricted range of communication channels, participants are often not aware of the presence of other participants or what they are currently doing; (ii) insufficient group coordination resulting from lacking or unfamiliar social protocols for virtual communication; (iii) deficient coherence of contributions, i.e., the chronological sequence of messages does not follow a coherent discourse logic or argumentation schema, and participants have trouble to infer for each message to which previous message(s) it refers. If learning discourses suffer from these problems, this poses a serious barrier to successful learning.

A remedy to overcome these problems might be to impose some structure on the communication and cooperation processes of learning groups. Two kinds of structured support for distributed cooperative learning can be distinguished: one is to provide support for representing knowledge, for example, to make computer-supported visualisation schemes available as in the Belvedere system (Suthers, 1999) where learners can co-construct conceptual graphs; another one is to provide support for the learning process itself, i.e., to use cooperation scripts to guide the communication and cooperation among participants. With respect to net-based cooperation, we distinguish two ways a cooperation script can be applied. One way is to provide the script via instruction, for example, presenting a written instruction which describes and illustrates how to execute the script, or to involve a human moderator who guides the discussion according to the rules of the script. A second way is to provide the script via implementation, which means that the rules of the script are implemented in the learning software and executed and controlled by the system during learning. Hron et al. (2000; Hesse & Hron, 1999) compared two kinds of instructional scripts (implicit: providing key questions to discuss, explicit: providing a set of rules for discussion participants were asked to follow) with a control group without script support, employing a text-based interface. They found a greater subject matter orientation for both structuring conditions and greater coherence for explicit structuring; however, knowledge acquisition was not affected by using a script. On a more microscopic level, an implemented script was studied by Hron et al. (1997). Dyads of learners discussed scientific problems using a combination of a text and a graphics interface. The script was related to the method of reciprocal teaching (Brown & Palincsar, 1989): in a strictly alternating sequence one learner had to provide an explanation which could be accepted or rejected by her or his partner; in case of rejection, an explanation had to be provided which could then be accepted or rejected by the other learner, respectively. Compared with a non-structured dialogue, the scripting had positive effects on problem performance and knowledge acquisition. On an even more fine-grained level, Baker and Lund (1997) and Soller et al. (1998) implemented an interface related to speech act theory that required learners to explicitly choose a pre-defined type of contribution from a set of communicative acts (e.g., question, justification, proposal). The types were indicated by sentence-openers such as “I propose to ...”, “to justify ...”, “I agree because ...”, which participants had to select and complete. They found increased task-focus and reflective communication when using such typed messages compared to free text communication.

What we call learning protocols are implemented types of scripted cooperation on a fairly microscopic level (Pfister et al., 1998). Different learning situations and different types of discourse require different learning protocols. We focus on learning protocols for synchronous, text-based discourse types for distributed small groups of learners, possibly including a tutor. In the next section, the notion of a learning protocol is defined in more detail, and two types of learning protocols are presented, one for explanation discourses and one for cooperative text processing discourses. Then, the interface and handling of an explanation protocol prototype is described. This protocol was applied in an experiment comparing the effects of learning protocol versus traditional chat with respect to knowledge acquisition; preliminary results of this experiment are reported and discussed.

LEARNING PROTOCOLS

As introduced above, a learning protocol is the implemented version of a specific discourse type relevant in cooperative learning situations. Such discourse types are, for example: explaining a concept or theory, disputing opposing points of view, joint construction of arguments, brainstorming, cooperative construction of content summaries, and many others. Any discourse type can be realized either by instruction or by implementation (via a learning protocol). Wessner and Pfister (2000; Wessner, Dawabi & Haake, submitted) introduced the notion of Points of Cooperation (PoC) as a general concept for the multitude of cooperation types significant for cooperative learning and proposed a framework on how to integrate Points of Cooperation into web-based courses.
Characteristics of Learning Protocols

In essence, a learning protocol requires the learners to make explicit what is usually implicitly handled in face-to-face discourse but poses serious problems in computer mediated discourse. The notion of a learning protocol as defined here is characterised by four features, each expected to solve one or more of the problems associated with non-supported net-based communication mentioned above:

(i) **Typed contributions**: Before sending a textual message, the type of message is explicitly defined by the learner. For example, if someone wants to explain something, she or he needs to indicate that the contribution is of type “explanation”; this type information is associated with the message and can be perceived by all participants of the discourse.

(ii) **Explicit reference**: For each contribution, the reference of the contribution is explicitly specified. A contribution might refer directly to a previous message, but it can as well refer to any other message or part of a message (a concept, an assertion, etc.). The reference is indicated in the interface (by an arrow) and can be directly perceived by all participants.

(iii) **Role assignment**: Each participant is assigned a definite role such as learner, tutor, or – more specifically – explainer, commenter, depending on discourse type. Roles and changes in role assignment are visible via the interface. Generally, a role defines the permitted types of contributions and is part of the rules determining turn-taking.

(iv) **Message sequencing**: The succession of contributions is defined and controlled by the system. Depending on the type of previous messages, the role of the contributing person, and a predefined sequencing scheme, the learning protocol establishes the actual sequence of messages; each participant is then informed whose turn is next.

Role assignment and message sequencing are intended to tackle the problem of social awareness. Message sequencing is also intended to support group coordination. The problem of deficient coherence is assumed to be ameliorated by typed contributions and especially by explicit referencing. Hence, we expect that communication guided by a learning protocol leads to a more coherent discourse structure, and, as a consequence, to a more coherent, more elaborated and more easily recollected mental representation of the knowledge to be acquired. The supporting function for learning of a learning protocol implies two aspects. First, it yields an increased quality of the ongoing synchronous discourse, and, second, it is useful as a mnemonic device, since saving the discourse generates a persistent learning object that can be replayed later and which is supposedly more helpful than saving and re-reading an unsupported chat protocol.

Communication and Cognitive Representation

On a theoretical level, we assume that there is a correspondence between the process structure of a discourse and the cognitive representation which is generated from carrying out the discourse. If the goal of a discourse is to exchange knowledge and, on an individual level, to acquire knowledge by mutually providing and receiving information, it follows that the structure of the emerging knowledge closely depends on the structure of the generating discourse. A structured and coherent discourse will generate coherent cognitive representations, whereas unstructured and incoherent discussions will lead to incoherent and fragmented representations.

From a psycholinguistic perspective, Clark and Brennan (1991; Clark, 1996) argue that the concept of grounding is essential for coherent and meaningful communication. Speakers need to make sure during the discourse that what they say is understood in order to construct a coherent sequence of utterances. Without common ground, i.e., without a certain amount of shared knowledge, communication is impossible (or extremely arduous). Clark and Brennan (1991) identify some basic methods for grounding in face-to-face communication, such as giving verbal feedback (e.g., “mmh”), initiating turn-taking (e.g., asking a question), or non-verbal attention cues (e.g., eye contact). These methods are not or only to some extent available in net-based communication media; in chat, for example, speakers can not rely on non-verbal cues to infer the understanding of others. Learning protocols, we assume, support the construction of common ground by requiring participants to explicitly provide the necessary information: referencing and typing of messages both function as grounding devices which increase the coherence of the discourse.

While Clark and Brennan (1991) focus on the analysis of observable communication acts, it remains unclear if there is corresponding cognitive structure. From a cognitive psychology perspective, script theory seeks to describe
the cognitive representations that determine the operation of prototypical event-action sequences (Schank & Abelson, 1977). Common learning discourses such as explaining and arguing are examples of communication scripts. We assume that experiencing or carrying out a communication script produces a memory trace, possibly some kind of “story” (Schank & Abelson, 1995), and that the memory trace is stronger the better the structure of the discourse matches the mental structure of the script. Cooperative learning from discourse means being able to recollect the structure of the original discourse, which is facilitated if the original discourse has been sufficiently coherent to be assimilated to the mental script structures.

In sum, we argue that the application of learning protocols has a twofold effect. On the one hand, learning protocols impose the necessary structure on discourse to generate coherence and common ground. At the same time, a coherent discourse is more easily assimilated to communication scripts and, consequently, better remembered. If this basic hypothesis turns out to be valid, learning protocols not only provide a pragmatic approach to support net-based learning, they also provide a theoretical approach to bridge the gap between communication theory and cognitive theory.

The Explanation Protocol
Currently, two implementations of learning protocols are under development. The first learning protocol aims at supporting explanation discourses, the second aims at supporting cooperative construction of text summaries.

The explanation protocol (EXP for short) is loosely based on the script employed by Hron et al. (1997). It supports distributed learning groups consisting of three to five learners and one tutor. It should be applied in situations when guided learning by a tutor (or, more generally, some kind of subject matter expert) is considered appropriate. The EXP-protocol instantiates a discourse of type “mutual explanation”, in which some concept or some limited domain of knowledge needs to be explained. Each member of the learning group is supposed to contribute knowledge by means of explanations, to ask questions to elicit explanations from other participants, and to comment (positively, negatively, or neutrally) the explanations and questions of others. The tutor plays a special role as the main source of explanations. As a result, each learner should reach a sufficient degree of knowledge about the topic under discussion, and the group as a whole should attain a high degree of shared knowledge.

In detail, the explanation protocol works like this (Figure 1):

1. The topic is introduced by presenting a short initial text that delineates the learning domain.
2. Each learner has to contribute a message in turn (floor-control is determined by the system).
3. A contribution is made by
   a. first indicating the reference, i.e., the learner marks the word, sentence or previous contribution which he or she refers to with the current contribution;
   b. second, the type of message is determined by selecting from a menu one of three message types: Question, Explanation, or Comment;
   c. then the message is actually written and

![Figure 1. Explanation Protocol Diagram.](image-url)
send to the public chat pane;

4. Depending on the message type, the next contributing participant is determined: if a question has been asked, the tutor is required to give an explanation; otherwise, if an explanation or a comment is delivered, the next learner is required to submit any kind of message.

This cycle continues until a specified time limit is reached or until all participants declare that they want to finish the discourse. The mechanism of the learning protocol can be depicted in a protocol diagram (Figure 1). There are three learners, consecutively numbered from one to three, and one tutor. Suppose it is learner one’s turn and she poses a question. Then, it is the tutor’s turn and she is required to give an explanation. If learner one would have given an explanation or a comment, then the floor would have switched to learner two. Now the tutor provides her explanation, and it is the turn of learner two. The same scheme now applies to learner two, and eventually the turn is back to learner one.

The Cooperative Text Processing Protocol

The learning protocol for cooperative text processing (COTEXT for short) is largely based on the work of O’Donnell and Dansereau (1992). It is intended for groups of three to five learners without any tutor or expert. A text, containing the knowledge to be acquired, is divided into several sections. For each section, one of the learners is assigned the role of a summarizer, and the other learners act as commentators. When a section is completed, roles switch and the next learner is assigned as summarizer. Summarising each section one after the other with changing roles constitutes the macro-level of the COTEXT-protocol. Within each section – the micro-level – the discussion is similar to the EXP-protocol. After providing an initial summary, each commentator contributes a message in turn. First, he references the part of the summary he wants to refer to, then selects the type of his contribution: a correction, a supplement, or a comment. If a correction or supplement is provided, it is the summarizer’s turn who can accept or reject the proposed contribution. If a comment is provided, it is the next commentator’s turn. This cycle repeats until no correction or supplement is given.

The COTEXT-protocol is an example of a learning protocol slightly more complex than the explanation protocol, since it consists of two levels: the macro-level controlling the sequence of sections to be summarized and the role switching, and the micro-level controlling the discourse about each single summary. In the next section, we describe the current prototype for the explanation protocol.

THE EXPLANATION PROTOCOL PROTOTYPE

The learning environment including the explanation protocol component was developed using the Dyce-Framework (see http://ipsi.gmd.de/concert/dyce). It is a server-client application for up to six distributed learners. Each learner and the tutor communicate via a client interface, which represents a chat enhanced with

![Figure 2. Explanation Protocol User Interface.](image-url)
typing as well as indicators about each participant’s role and turn-taking (sequencing). Figure 2 depicts the user interface including markers for the protocol functionalities (these markers are of course not visible for the users). The reference of a message is indicated by an arrow; in Figure 2 the reference is just to the previous message, but it could as well go to any previous message, or to any component of a previous message (word, sentence), or to any component of the initial text on the left pane.

The middle pane shows the ongoing chat, i.e., the progression of contributions (each contribution associated with the contributing participant and the message type), and the input field for the current message. The menu containing the three available message types comment, question, and explanation is also visible. The right upper pane depicts the roles of the participants, and the participant who is currently active is emphasized. The lower right pane is for general instructions.

A learning protocol session starts with an empty chat and the presentation of the initial text in the left pane; this text is always visible during the ensuing discourse. One learner is randomly selected to begin and he or she is now required to indicate the referred to component (at the beginning, this will be some concept or sentence of the initial text), select the appropriate type and write the message. After pushing the send button the message becomes visible for all participants, and the learner whose turn is next is indicated; if a question was asked, it is the tutor who is next.

**COMPARING LEARNING PROTOCOLS WITH UNSTRUCTURED CHAT**

As a preliminary base-line test of the efficiency of learning protocols we compared an explanation discourse guided by the EXP-protocol with an equivalent discourse using conventional free-text chat as a control condition. For this comparison it is immaterial if net-based learning is superior to face-to-face learning or if cooperative learning is superior to individual learning (Dillenbourg, 1999; Dillenbourg et al., 1995). The main independent variable was learning with versus learning without the explanation learning protocol. As a second independent variable, the type of knowledge to be acquired was varied: one domain was “earthquakes” and the second domain was a philosophical topic on the epistemological concepts “knowing and believing”. We expected that the learning protocol was especially efficient when applied to a scientific domain such as earthquakes with unambiguous definitions and causal relations, in contrast to a philosophical domain with more vague and imprecise concepts.

**Method**

A total of 24 subjects (students of the Universities of Tübingen and Darmstadt) participated in the study, put together in groups of three, i.e., eight groups. Participants received 30 Deutsche Mark for participation in a two-hour session.

The learners and the tutor worked with standard PCs in isolated cubicles (i.e., they could not see each other), simulating a distributed scenario. First, the general scenario was introduced, a short questionnaire about experience with computers and the internet was administered, and the user interface was explained. In the experimental condition, participants learned how to refer to previous messages, how to choose a message type and how the sequence of messages was indicated. In the control condition, which consisted only of a standard chat interface, they only learned how to enter messages. Then, participants worked through both knowledge domains (earthquake, philosophy) successively, with a short break in between. For each domain, a learning goal was provided and presented on a sheet of paper attached to the PC. The learning goal for the earthquake domain was “to understand the causes and consequences of earthquakes and the different types of earthquakes”, for the philosophy domain the goal was “to understand the meaning of the concepts knowing and believing, and differences and similarities to related concepts”. The time limit for a learning session was 25 minutes. Directly following each learning session, a knowledge test was applied to assess participants’ degree of knowledge after learning; the knowledge test consisted of a number of multiple choice items and one open question.

In the experimental condition (with explanation learning protocol), the protocol largely controlled the session according to the rules explained above. Whenever the tutor was required to give an explanation, he or she provided an answer taken from a standardized set of answers. These standardized answers consisted of short explanations of key concepts (such as epicenter or seismic waves for the earthquake domain). If the question contained a key concept, the standardized explanation was given, otherwise, the tutor answered “sorry, I do not know the answer”. In the control condition (free text chat) participants were free to enter any kind of message at any time. The tutor
monitored the messages and whenever a message could be identified as a question containing a key concept the standardized answer was given; else, the tutor did not join the discussion.

Results
The effect of the learning protocol was tested with the test score of the knowledge tests as the dependent variable. The score for the earthquake knowledge test could range from 0 to 17 points, for the philosophy test the range was from 0 to 15 points. The mean scores for the earthquake domain were 12.08 (SD = 2.65) for the experimental condition, and 8.67 (SD = 1.93) for the control condition. As expected, participants in the experimental condition learned significantly more than participants in the control condition, t(22) = 3.61, p < .01. The mean test scores for the philosophy domain were 9.04 (SD = 1.59) for the experimental condition, and 8.21 (SD = 1.54) for the control condition. Though participants in the experimental condition learned slightly more than participants in the control condition, the difference turned out to be not significant.

DISCUSSION
As a caveat note that the results are preliminary based on a rather small sample (N = 24); also, only a post-test was applied. We are currently running experiments using a larger sample and a pre-post-test design to assess differential increases in knowledge, additionally, the cooperative text processing protocol is tested. Further evaluations related to the learning protocol approach are currently conducted in cooperation with other project groups (Holmer & Holst, submitted; Wessner et al., submitted).

However, the positive effect of the learning protocol in the earthquake domain is quite strong, and in the philosophy domain the difference is in the expected direction. This confirms the potential of learning protocols to enhance immediate knowledge gains from net-based discussions. It also shows that the efficiency of learning protocols might heavily depend on the type of knowledge domain under discussion.

However, several questions remain and need to be studied further. Given the effect of learning protocols for cooperative learning at least in some knowledge domains, the question is: which features of the learning protocol are essential, and which are optional or even hindering knowledge acquisition? So far, it is the joint working of referencing, typing, sequencing and role assignment that leads to improved learning. We need to separate out which characteristic is important under which conditions. It could be assumed that message sequencing and referencing are mainly responsible. Message sequencing not only imposes an explicit order on turn-taking, it also implies an equal distribution of messages across participants. A common finding in face-to-face discussions as well as in net-based chats is the imbalanced amount of contributions delivered from participants; generally, a few learners dominate the discourse, and others hardly ever contribute anything (be it because they are too shy or because the take the opportunity for social loafing). In the learning protocol condition, all learners were required to contribute when it was their turn. Assuming that superior learning results from active contributing to a common discourse, the average amount of learning will be higher under a learning protocol.

The need to indicate the referent of each message might have two positive effects. During message construction, the message is necessarily closer related to the ongoing discourse, which should increase its elaboration for the learner who delivers the message. For the recipients of the message, interpretation is greatly facilitated, since they need not actively and effortfully search for the intended reference. Hence, intended reference by the speaker and perceived reference by the recipient are identical and common ground is directly established. The resulting coherence of the ongoing discourse will then improve understanding and recollection of shared knowledge. If this corresponds to a more coherent cognitive representation, as discussed above, is an open question. Both characteristics, referencing and sequencing, are investigated in our ongoing research.

The role of message typing is less clear. From the tutor’s perspective, typing disambiguates the messages and makes it clear when a question is asked. In the control condition (untyped chat), there was some leeway in the interpretation of messages, i.e., if a question was asked or if a learner only wanted to show some doubt or comment on something. A lenient tutor would provide more information than actually required, whereas a more rigid tutor would withhold useful information. From the learners’ perspective, typing requires to make a decision about the message before the message is actually created; also, any message was subsumed under only three broad categories. A detailed inspection of the learning sessions shows that explanation are of many kinds, for example, definitions, rephrasings, or causal relations (and many more). To include such sub-types as predefined types, however, would
make the task for the participants even more effortful and it would be rather uncertain if the learners’ interpretation of types coincides with the interpretations of the other members of the learning group.

Not entirely unexpected is the finding that learning protocols work better in some knowledge domains than in others. A learning protocol such as the explanation protocol particularly supports the acquisition of facts, or declarative knowledge. Philosophical domains involve much more negotiation about vague concepts (as are the epistemic concepts used in the experiment). One might conclude that the explanation protocol is better suited for declarative knowledge acquisition in science or technology oriented domains, whereas the cooperative text processing protocol is better suited for philosophical or ethical domains.

There is a further ambiguity concerning the measurement of learning effects by means of a standard knowledge test. Note that the learning protocol does not imply that the subject matter to be learned is actually covered and discussed exhaustively; the protocol only provides constraints on the procedural aspects of the discourse, but is completely ignorant towards the content. Hence, there is no guarantee that the intended knowledge domain as defined by the set of knowledge test items coincides with the domain actually discussed during the learning session. It could either be the case that most of the domain was discussed, but in a rather shallow way, or that only a portion of the domain was discussed, but in a deep and reflective fashion. To clarify this issue, a detailed qualitative analysis of the learning sessions needs to be carried out.

In sum, we argue that learning protocols are a valuable tool for a multitude of cooperative learning situations. The conditions for the efficient application of learning protocols, for example, which content domains or which situational contexts are appropriate, need further clarification. Note, however, that we do not propose to establish learning protocols as the standard way of cooperative learning. From a motivational point of view, this would probably be unacceptable. Also, individual differences and preferences are much too large to be put under a limited set of standard protocols. What we recommend is to use learning protocols as a kind of short time exercise with clearly defined objectives and time restrictions. Metaphorically speaking, a learner (or group of learners) could use learning protocols in a similar manner as an athlete (or a team) uses special exercise procedures such as lifting weights or making push ups. Normally, free play is much more fun than constrained exercises, as free chatting and discussion is more fun than controlled discourse. However, selecting the right exercise, i.e., the appropriate protocol, at the right time for the right purpose, might enhance performance in a highly efficient way and normal discussions might benefit from this experience.

ACKNOWLEDGMENTS
This research is funded by the German Research Council (DFG), grant PF330/1-1, as part of the priority program “Net-based knowledge communication in groups”. Many thanks go to Martin Wessner, Mareike Hennings and Werner Müller for valuable comments and for conducting the experiments.

REFERENCES


Wessner, M., Dawabi, P., & Haake, J. (submitted to CSCL02). The power of PoCs: Moving from individual to collaborative learnflow.