A learner support model based on peer tutor selection

Peter van Rosmalen, Peter Sloep, Liesbeth Kester, Francis Brouns, Marcel de Croock, Kees Pannekeet and Rob Koper

All authors are employed at the Open University of the Netherlands.
Address for correspondence: Peter van Rosmalen, Open University of the Netherlands, P.O. Box 2960, 6401 DL Heerlen, The Netherlands.

Email: {Peter.vanRosmalen, Peter.Sloep, Liesbeth.Kester, Francis.Brouns, Marcel.deCroock, Kees.Pannekeet and Rob.Koper} @ou.nl.

Abstract
The introduction of elearning often leads to an increase in the time staff spends on tutoring. To alleviate the workload of staff tutors, we developed a model for organizing and supporting learner related interactions in elearning systems. It makes use of the knowledge and experience of peers and builds on the assumption that (lifelong) learners, when instructed and assisted carefully, should be able to assist each other. The model operates at two levels. At level 1, prospective peer tutors are identified, based on a combination of workload and competency indicators. At level 2 the thus identified prospective peer tutors become the actual tutors; this is done by empowering them with tools and guidelines for the task at hand. The paper will situate the model in networks for lifelong learning. For one kind of interactions, answering content related questions, we will review a set of existing approaches and emerging technologies and describe our model. Finally, we will describe and discuss the results of a simulation of a prototype of the model and discuss to what extent it matches our requirements.

Keywords
Elearning, Lifelong learners, Tutor workload, Peer support, Latent Semantic Analysis.

Introduction
The introduction of elearning often leads to an increase in the time staff spends on tutoring (Bartolic-Zlomislic & Bates, 1999; Bacsich & Ash, 2000; Koper, 2004). This occurs because often an extended classroom model is followed: a teacher would lecture as usual and keep regular office hours. In addition to this, he/she would typically create a website to support the course and be available for email help between classes. Part of the answer to this problem is to move away from an extended classroom model and adopt a distributed learning approach (Ellis et al., 1999).

Networks for Lifelong Learning ('Learning Networks') exemplify such a distributed approach. A Learning Network (Koper et al., 2005; Koper, 2006) is a self-organized, distributed system, designed to facilitate lifelong learning in a particular knowledge domain. A Learning Network is specific to a certain domain of knowledge (e.g. an occupation) and consists of:

a) Lifelong learners (Learning Network users): people with the intent to learn and the willingness to share their knowledge in the specified domain.
b) Activity Nodes: collections of learning activities that are created and shared in order to exchange knowledge and experience, or to develop competences in the domain.
c) A set of defined learning outcomes, or 'goals' (e.g. competence levels).

But even in a Learning Network's approach it remains necessary critically to look at the time staff requires to support students:

- Learners likely do not arrive in groups, nor have the same objectives or background. The heterogeneity of the group of learners and the lack of a readily available social structure to give mutual support make large demands on staff tutors. In an online learning context (Anderson, 2004) staff can no longer assume well-defined and pre-planned tasks but have to adapt to student needs on the fly.
- The accessibility of staff tutors by email makes online learners expect a quick answer to emails they have sent (Salmon, 2000); even worse, they expect personalised answers.

As a consequence, also for a Learning Network a model is needed that details how to organize and support the learners. One characteristic of Learning Networks makes the need for a support model even more urgent. A Learning Network does not merely focus on formal learning but also aims to support non-formal learning. In such cases, no staff at all may be available. And yet, also here, learners will want to know e.g. how to proceed or how to understand the available Activity Nodes.

Support activities
A brainstorm session (De Vries et al., 2005) with a group of stakeholders identified four groups of 'critical' student support activities. They are critical in that they easily lead to staff work overload. The four groups are:

- *Assessment of student contributions*: in particular, to give formative feedback and to detect plagiarism.
- *Answering questions of students:* to route questions to the appropriate person and to formulate a personalised answer.
- *Monitoring and assessment of study progress*: ranging from dropout prevention to providing personalised advice.

- *Community and group support*: to select and create groups, to order and archive threads, to provide overviews of the activities of a community as a whole and of the individual actors.

We chose first to concentrate on answering questions because:

- Question-and-answering involves continuous interactions and consequently can be very disruptive for staff.
- Learning may improve when students can ask questions and subsequently receive relevant answers. Few learning environments offer students the opportunities and facilities to ask questions and receive answers (Howell, 2003).

Support activities in a Learning Network

In this article we propose a support model that automatically invokes peer learners to give support. Suppose we have a Learning Network in domain *D*, e.g. psychology, with a set of Activity Nodes (AN) A_1-A_{10} (Figure 1). Moreover, we have a Lifelong Learner *P* (Paul) who has formulated a goal that can be achieved by studying A_1, A_2, A_3, A_6, A_7, A_9 and A_{10}. Next, we know that Paul, in view of his working experience and prior studies, has exemptions for A_5 and A_6 and has already successfully finished A_7. Finally, let's assume that Paul while studying A_1 runs into problems. He has a problem understanding the relations between a number of concepts and as a consequence he is not able to complete an assignment. He studies some additional literature and searches the web, to no avail. Paul decides to pose a question; he describes the general problem and his question.

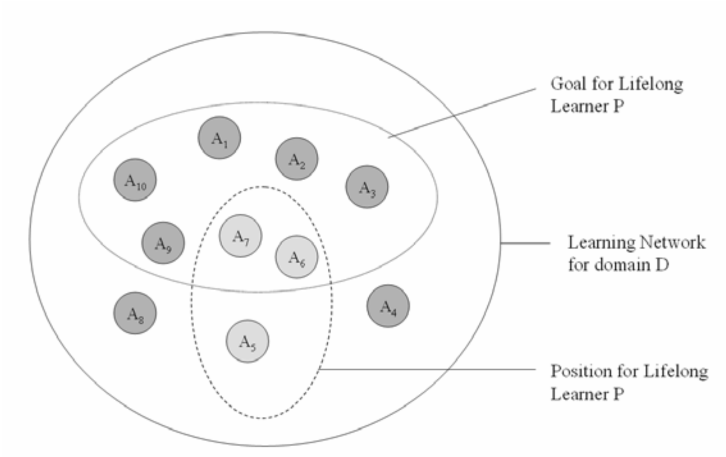

Figure 1. A Learning Network for domain *D*.

This scenario suggests various requirements for our support model. We will discuss these now more formally and then move on to review existing approaches and emerging technologies that might help meet these requirements.

However, before doing so, we should point out that the present paper is part of a series of articles. Koper et al. (2005) set the stage by defining the context, that of a Learning Network. De Vries et al. (2005) identified the needs, as just discussed. Kester et al. (in press) described the model from an educational, pedagogical and community perspective. Van Rosmalen et al. (2006) focused on the usage of Latent Semantic Analysis, the (required) calibration approach, its result, and a simulation. In these articles little attention has been paid to what technologies exist to implement the question-answering model we seek to develop. The current manuscript tries further to elaborate the picture by articulating requirements, reviewing existing approaches, and - underpinned by these findings - detailing a model.

Requirements

We distinguish four types of requirements: quality, involvement, empowerment and portability.

- *The model has to* ***alleviate*** *the support task for the staff tutor while maintaining quality.* It means that (part of) the answering is done without staff intervening and that the answer has to meet a minimum quality level. Thus the model should increase the number of students a staff tutor can support. Wiley (2004) captures this challenge in one concept: the teacher bandwidth, the number of students a teacher can serve in distance education.

- *The model has to **involve** a substantial fraction of the members of a Learning Network community and make optimal use of their knowledge.* A Learning Network as a self-organized, distributed system depends for its functioning on the learners' willingness and time to share their knowledge. If only a small portion of the learners actually contribute answers they themselves now may become overloaded or there will be little sharing of knowledge. Equally important, supporting each other on a topic just mastered can be a valuable experience (for a detailed discussion on the underlying theoretical aspects of our model on learning in communities and peer tutoring see: Kester et al., in press). Providing peer support may strengthen the social relations and can help achieve better learning outcomes (Fantuzzo et al., 1989). In particular lifelong learners can, given their experience, easily change roles from student to coach and move between learning and working (Anderson, 2004). Obviously, we have to acknowledge the time constraints of lifelong learners. Therefore the model should be able to involve competent peers while at the same time evenly spreading the workload.
- *The model should be able to **support** the selected actor in performing the task at hand.* A clear support structure is beneficial to the quality of the support task, if necessary it may even contain a quality control loop. The structure should also allow the learners to concentrate on the content of the task; this benefits their learning outcomes. For the current case it implies that we are looking into how learners can help each other answering a question.
- *Finally, the model should be **portable**.* The model proposed should not require extensive domain dependent tuning, preferably none at all. In the same vein, the implementation of the model should not be system dependent. It should be relatively straightforward to add the model to any virtual learning environment by building on a combination of learning technology standards and technical interface standards.

Existing solutions

A wide choice of solutions exists for the task selected, answering content related questions, ranging from groupware, helpdesks to virtual assistants. We will discuss each of them paying special attention to an example of language technology i.e. Latent Semantic Analysis (LSA). Question-answering depends on an understanding of natural language. The use of language technology may enable us partially to automate question-answering. LSA has been used already in a variety of educational settings, such as essay grading and question-answering.

Caron (1999) gives a broad overview of groupware systems. They range from general purpose, pre-web technology Usenet discussion groups; via dedicated question-answer systems intended to solve problems building on a combination of posting and brokering; to still popular recommender systems such as Slashdot (www.slashdot.org). Two of his findings are of interest here. Often there is a small group of users who 'altruistically' reply to contributions. Thus on the whole, only a small number of participants is responsible for a large percentage of the contributions. This makes the use of groupware rather unpredictable and hence unreliable, unless there is a facilitator or a high number of users. Similar conclusions have been drawn in educational settings (Guzdial, 1997; Anderson, 2004). Both Guzdial and Anderson underline that if participation is desired, there should be clear incentives and guidelines. This seems true in particular for lifelong learners. They participate in many activities that compete for their time, and thus need convincing arguments to join in yet another activity.

Helpdesks (Woudstra et al., 2004) are another common solution to deal with questions. A helpdesk is often used as a first-line aid, or as a means to forward a question to an appropriate person in the organisation. Ideally, a helpdesk learns from previously asked questions and it accumulates relevant data on its customers. A helpdesk therefore requires staff tutors but only if the type of question requires their expertise or their formal involvement. A successful helpdesk should quickly pay back its investment. Unfortunately, in our case a substantial number of the questions learners will pose is directly related to the content of the activities they are involved in. Given the broad coverage of topics a Learning Network is expected to deal with, it will be difficult to staff a helpdesk adequately and yet avoid running into the teacher bandwidth problem.

Another way of helping customers with their questions, separately or in combination with helpdesks, is to create a FAQ (Frequently Asked Questions) or online virtual assistant. There is a fast growing number of virtual assistants in all areas of business (see e.g.: mysiteagent.com, www.nominotechnologies.com). They apply a combination of agent and language technologies and operate not only via the web, but also via instant messaging or cell phones. At the EDUCAUSE 2003 conference (Gaston, 2003) an example of such an assistant was presented that allowed students to ask questions such as "when do classes start". Though useful if created carefully they are insufficient if they operate on their own because it will be too difficult and time consuming for them to offer sufficient coverage. Other more general examples of agents are I-Help (Vassileva et al., 2001), Yenta (Foner, 1997) and Expertfinder (Vivacqua & Lieberman, 2000). They do not rely on a set of pre-designed question-answer pairs but, based on a set of characteristics, try to find a suitable person or, as in the case of I-Help, a suitable person or material.

I-Help is based on a multi-agent architecture, consisting of personal agents (of human users) and application agents (of software applications). Each agent manages specific resources of the entity it represents, including for example knowledge resources or instructional materials. If a user requests help, the agents communicate with each other and with matchmaker agents to identify appropriate help resources. If an electronic resource is found (represented by application agents), the personal agent "borrows" the resource and presents it to the user in a browser. However, if a person is identified, the agents negotiate the price for help, since human help involves inherent costs (time and effort) for the helper. Help is arranged (negotiated) entirely by the personal agents, thus freeing the users from the need to bargain. In this way the personal agents trade the help of their users on a virtual help market.
Yenta, a multi-agent matchmaker system has been designed to find people with similar interests and introduce them to each other. Yenta seeks to assist people in finding people with relevant expertise. It does so by involving the majority of "lurking" people instead of turning to those people who are already active. Yenta assumes that that two users have similar interest if both possess similar documents (emails, newsgroup' articles, files).
Expertfinder is an agent that classifies novice and expert knowledge by analysing documents created while working in the domain of Java programming. The user models are automatically generated and allow for matching of a novice's query to an appropriate expert. The system tries to distribute the workload evenly when more experts are available. It also does not prioritise the best expert but someone whose knowledge level is close to the questioner's. This way, it is more likely to bring together people who share a similar mental model of the problem discussed. The number of success cases reported, i.e. experts able to find an answer, was 85%. Interestingly, in 50% of the cases, the expert was able to give an answer only after looking it up.

Latent Semantic Analysis
Question-answering depends on understanding natural language. Therefore it is worthwhile to consider the use of language technologies. They may help us automate question-answering, if only in part. An example of particular interest because of its widespread use in educational settings is *Latent Semantic Analysis* (LSA) (Landauer et al., 1998; Van Bruggen et al., 2004; for a brief technical introduction to LSA see http://research.nitle.org/lsi/lsa_definition.htm). LSA has its roots in research on document retrieval. LSA connects related words in a number of steps (e.g. in documents on Computer Science the words human, computer and interface are related). In this way, although the actual keywords in documents may differ, LSA may show them to be associated through these kinds of semantic similarities. By relying on measures of semantic similarities between documents, LSA is able to improve retrieval beyond keyword matching (Dumais, 2003). Among other things, LSA has been used extensively and successfully for automated essay grading (Foltz et al., 1999), in intelligent tutoring environments (Graesser et al., 2000) and to help answer questions. HURAA (Person et al., 2001) and FAQO (Caron, 2000) are examples of systems in which the user can ask questions formulated in natural language.
HURAA is a web-based information delivery and retrieval system that guides the user through six distinct learning trajectories. At any point during a learning session the user may ask a question. The question is mapped into an LSA text space built of a variety of documents plus a corpus of question-answer pairs. LSA is used to locate the five best text segments for the user. FAQO is a (prototype) system that allows the users to query questions in natural language in order to find relevant documents to solve their problems for specific technical problems. The objective of the system is to support the staff involved in answering these questions. The system constructs an LSA text space from email archives and other existing documents in the problem area concerned. LSA is then used for query matching.

Summarizing the various approaches
All examples discussed deal with answering questions. Looking at the way in which the answers are given one can distinguish three types of approaches. The first relies on stored answers (helpdesks, FAQ, virtual assistants), helpdesks are included because of the limited capability of their staff to answer not-anticipated questions. The second approach relies on finding the right person to answer.

The person can be loosely coupled as with groupware. Here the poser of the question just has to wait until someone volunteers. Alternatively, a person is carefully identified as in the agent-based systems (I-Help, Yenta, Expertfinder). In the third approach (a contribution to) an answer is automatically identified with the help of LSA from a corpus of documents built from the topic under discussion. The first approach does not fit our requirements. It relies on a labour-intensive preparation of possible answers for each domain and in many cases it will still need staff to assist. The second approach, however, seems to fit the bill, even more so if we combine it with the third approach. LSA can be used to assist in identifying relevant documents to answer questions formulated in natural language. The resulting documents can then be used to assist the persons identified in giving an answer. This combination of carefully selected persons *and* documents we will therefore adopt to develop our own support model.

The model: alleviating the tutor load

Broadly speaking, the model describes how to select and support a group of lifelong learners that will help to answer a question of one of their peers. The staff tutor will only interfere if triggered, for example because an answer is not in time or does not meet a pre-specified minimum quality rating. Staff may also interfere of their own volition, for instance to assure the quality over time by sampling answers regularly. The model addresses both the need of learners to receive personalised, individual feedback and the need of staff tutors to keep their workload within bounds. It makes use of the knowledge and experience of peer learners. It builds on the assumption that lifelong learners, when instructed and assisted carefully, should be capable to assist each other e.g. in carrying out joint assignments, giving peer-assessments or answering question of each other. The model distinguishes four types of participants (Figure 2):

- a learner (tutee) who asks for support;
- a learner who acts as peer tutor and provides support;
- for every learner, a Personal Agent (PA) that assists in maintaining his or her data;
- a Match Maker Agent (MMA) to organize and control the interactions between the actors (learners and their personal agents). Both the PAs and the MMA will consist of a set of specialised agents which deal with specific tasks, e.g. an agent that proposes pieces of text suited to help answering the question.

The model builds on the assumption that learners have been registered and that their ‘position’, the combination of successfully completed Activity Nodes (ANs) and the ANs they have exemptions for, is known. The model assumes that learners know the contents of an AN if their position includes the AN in question.

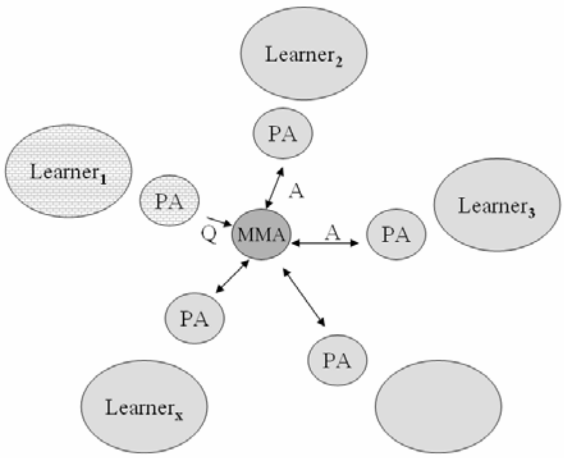

Figure 2. Schematic drawing of asking a question: (1) Learner$_1$ poses a question. (2) The Match Maker Agent selects and negotiates with the Personal agents. (3) Learner$_2$ and Learner$_3$ supply an answer.

The approach followed contrasts with other approaches in which people are appointed beforehand (tutors, outside experts or peers from the same class). In Learning Networks, in general, there are no classes and people will have a variety of backgrounds and study plans. Hence the group is created ‘on

demand' and expected to exist only for as long as is required to support the request. Clearly, although this 'ad hoc' community itself will be transient, the relations that have been forged during its existence may last. Indeed, it is hoped that they will thus be establishing a higher degree of self-organization of the Learning Network.
The model recognises five main steps. In the first three steps the working context is defined. The steps are *creating a request*, *defining its context*, *identifying suitable candidate peer tutors*. In the last two steps the actual request for support is addressed (*creating the answers*) and the question poser (tutee) passes judgment on the answer and the contributors (*the tutee receives the answer*). The assistance of the staff tutor is required only if a question is not successfully resolved or if a learner (repeatedly) is refusing to participate or is rated poorly.

Creating a request. The learner who intends to ask a question, will receive a form with guidelines and a request for additional info, e.g. on the urgency of the question. We have decided to restrict the model to content related questions. The learner receives instructions that technical questions (e.g. "I cannot access the content. What to do") or procedural questions ("when and where can I do my examination on …") are considered to be out of scope and should be asked elsewhere.

Defining the context of the request. Usually, the question asked will be related to the AN the learner is studying at that moment. This need not be the case, though, the learner might study more ANs at the same time and there could be other ANs that relate to the question. Therefore this step determines the ANs containing information that is relevant to the question. In a way similar to Yenta we look at the similarity of documents. We use LSA to calculate the similarity between the question and the documents of the ANs. The ANs that best fit (a combination of the number of documents that have a high similarity and the level of similarity) the question are considered relevant.

Identifying suitable candidate peer tutors. The next step is to find and select, based on the context defined, suitable peer tutors and to decide on the optimal number of peer tutors. The community that thus arises should be large enough to guarantee that an answer becomes readily available but small enough to minimize the chance of duplication of efforts. Obviously, what the optimal size is cannot be decided a priori, it is an empirical question. A size of 1 could in principle suffice, but this one person may not be available or may give an inadequate answer; the entire Learning Network would maximize the chance of a quick answer, but such a strategy is bound to lead to duplication of efforts. Also, too large a community would dramatically increase the number of lurkers. About five seems to be adequate (Kester et al, in press). The system now attempts to form such an ad hoc and transient community by inviting learners who, according to four different criteria, are most suited to answer the question (see Table 1 for the selection formula). The suitability ranking is a weighted sum of *tutor competency*, *content competency availability* and *eligibility*:

- The *tutor competency* (T_L) is the ability of a peer learner to act as a tutor. The tutor competency is derived from a combination of data logging, i.e. from the frequency and size of the contributions, and ratings on answers given previously.
- The *content competency* (C_L) indicates if a learner has successfully finished the ANs related to the question; more precisely, it is the weighted sum of the status of all relevant ANs. A more sensitive measure could be obtained by weighting the ANs according to the time elapsed since their completion: the more recent, the larger the weight.
- *Availability* (A_L) is based on the actual availability as derived from the personal calendar of the learners and their past workload. This measure is time-dependent: recent workloads should affect availability more than ancient workloads.
- Finally, *eligibility* (E_L) measures the similarity of the learners. It looks at which other ANs, outside the question specific ANs, the potential peer tutor and the tutee have in common. There are two reasons to use this measure. Some learners will have more expertise than others. The total tutoring load is therefore likely to increase rapidly with increasing expertise. However, an unequal spread of the tutoring load is undesirable. Learners should only spend limited time and effort on tutoring. By considering similarly advanced learners only, one avoids piling up questions on the advanced students. There is an additional, pedagogical twist to this argument. If tutoring is an educationally valuable experience *per se* - and not just a matter of community service - then learners should act as tutors for learners with a similar not too distant expertise level and background to achieve higher learning outcomes themselves. The eligibility of a learner guarantees that 'near-experts' (near in the meaning of having expertise close to the user asking the question) are prioritised.

Table 1. The main formula to select peer tutors and the parameter setting applied.

Explanation	Formula	Parameter setting
Tutor suitability of learner L: Ts_L. A number between 0 (not suitable at all) and 1 (very suitable). Parameters WT,WE,WA,WC to adjust the relative importance of the four factors. **Notes**: (1) to assure a minimum level of knowledge, the four factors are *only* calculated if the Content competency > 0. (2) to assure presence, if available time in the question period is zero the learner in question is removed from the list.	$Ts_L = ((WT \times T_L) + (WE \times E_L) + (WA \times A_L) + (WC \times C_L)) / (WT + WE + WA + WC)$	WT = 0 WE = 0.5 WA = 0.5 WC = 1
Tutor competency: T_L. A number between 0 and 1. Parameters Tw_1 and Tw_2 to adjust the relative importance of Te (on average how active the learner behaved in previous questions) and Tr (on average how previous answer were rated).	$T_L = ((Tw_1 \times Te) + (Tw_2 \times Tr)) / (Tw_1 + Tw_2)$ with	n.a. (WT = 0)
Eligibility: E_L. A number between 0 and 1. E_L is taken relative to Lq, the learner who asked the question. It is calculated over all ANs that do not relate to the question.	$E_L = (\text{Sum}_{(i=i,2,..,N\ \&\ \text{all } i \mid ANi \text{ is not question related})} (\text{score}(ANi_L) = \text{score}(ANi_{Lq})))/(N - \#\text{ question related } ANi\text{'s})$	The score of AN can be 0 (not started), 0.3 (started), 1 (assessment completed successfully).
Availability: A_L. A number between 0 and 1. Parameters M (max_extra_workload) and Tp (timeperiod over which the workload is calculated). The availability depends on the relative past workload. It compares the number of times a learner is involved in answering a question relative to the other learners in a given time period.	A_L = one of {0,.0.25, 0.5, 0.75, 1}. The value is 0.5 if L has contributed on average; 0.25 if L has contibuted above average but no more than M above average; 0 if L has contibuted more than M above average etc…	M = 1
Content competency: C_L. A number between 0 and 1. Parameter D to adjust the number of documents to calculate correlations for. Dt is the number of text fragments offered. W_{ANi} is based on the correlation between the question and the documents in ANi. The correlation is calculated with LSA. C_{ANi} is the Content competency for ANi. **Note**: The value of C_{ANi} takes into account the score, the time expired since completion and the study time of the ANi.	$C_L = (W_{AN1} \times C_{AN1}) + (W_{AN2} \times C_{AN2}) + \ldots + (W_{ANn} \times C_{ANn}) / (W_{AN1} + W_{AN2} + \ldots + W_{ANn})$	D = 3 Dt = 3 Note: C_{ANi} only based on the score of AN

Supporting and creating the answers. Based on the suitability ranking above, a number of learners are invited to join a wiki and assist in answering the question. The invitation includes the question, guidelines and a small set of documents (or paragraphs thereof) that have been identified as relevant to drafting an answer. The guidelines *and* the documents together form a support structure for the invited peer tutors. The documents are derived with the help of LSA, in a similar way as explained before. The objective is to help the peer tutors to get a quick overview of documents relevant to the question.

The tutee receives the answer. After some time, the peer tutoring process ends and a response becomes available. Ideally, the process ends because the tutee is satisfied with the answer. However, if this is not the case, it may also end because a predefined period of time has elapsed or because the learners agree to end it. Whatever the reason, the tutee should rate the work of the peer tutors by rating their collective answer. If necessary, these data are used, to alert a staff tutor that there is an unresolved question or (in combination with other logging data) that some learners do not perform as peer tutors as required.

A first simulation

To test our model we decided to build a prototype. We used a server-based architecture since, in this way, most of the required components (Figure 3) were readily available. To assure that the prototype is viable we calibrated the LSA-parameters, and simulated and tested two key aspects. First, we checked how well we can use LSA to identify the topic of a question (i.e. to which AN(s) a question belongs) and to select text fragments useful for answering the question (Van Rosmalen et al., 2006). Second, we checked if the peer selection formula met our expectations.

The prototype

The prototype (Figure 3) consists of five modules. The learners will only notice a Learning Network, its ANs and a question-interface; additionally, for each question there is a wiki that includes the question and three documents selected from the Learning Network's ANs. All are implemented in Moodle (www.moodle.org). The wiki is populated with both the tutee and the learners who accepted the invitation to help (the peer tutors). For the designer and for the runtime system we have three additional modules: a General Text Parser (GTP; Giles, Wo & Berry, 2001), a GTP calibrator (GTP Usability Prototype (GUP); De Jong et al., 2006) and a tutor locator (ASA Tutor Locator (ATL); Brouwers et al., 2006). We use GTP, an LSA implementation, to map the questions on the documents in the Learning Network. The GTP module returns correlations between the question and documents. The correlations are used to determine the AN to which a question fits best and to select relevant text documents. The GUP module supports the calibration of the LSA parameters. Finally, the ATL module finds and invites the peer tutors.

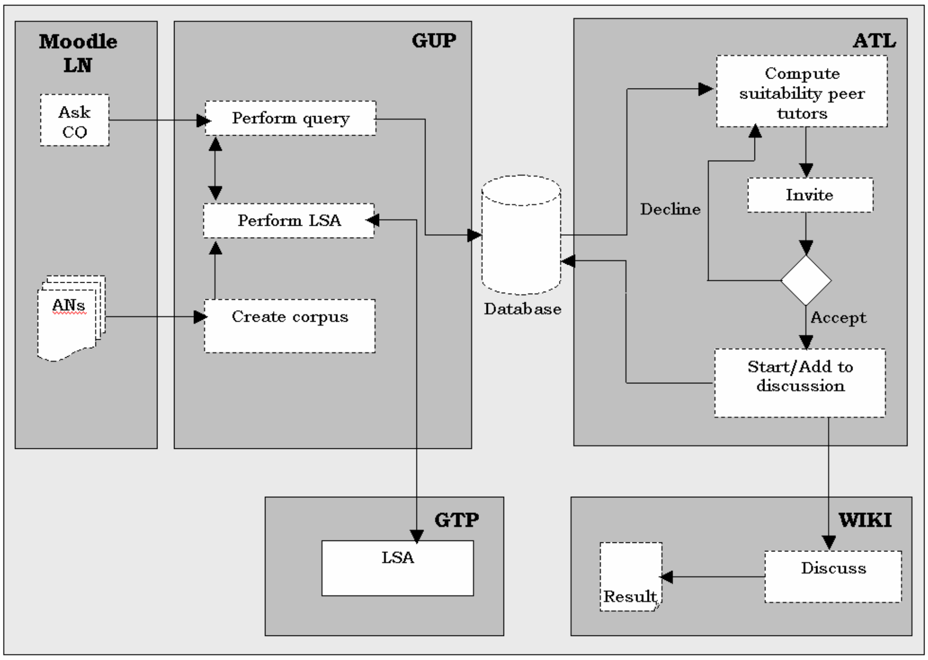

Figure 3. The main modules of the prototype.

AN identification and text fragment selection

For the simulation we used an existing Learning Network, the domain of which is basic internet skills (Janssen et al., in press). It contained 11 ANs, each of which introduced a different aspect of the Internet and consisted of an introduction, exercises, references to external web pages for further study, and an assessment. The Learning Network matches our two initial requirements i.e. (1) the text corpus could be accessed (a combination of Moodle and external web pages) and (2) the users' progress could be tracked (by the data available from the AN assessments). We formulated a set of 16 test questions, each related to exactly one AN. For each question, the prototype proposed three text fragments as well as determined the source AN.

Table 2. Position of learners L_1-L_5 for the selected Activity Nodes

	L_1	L_2	L_3	L_4	L_5
Score AN1	1	1	0.3	0	0
Score AN2	0.3	1	1	0	0
Score AN3	0	0.3	1	1	0.3
Score AN9	0	0	0	1	1
Score AN10	0	0	0	0	0
Score AN11	0.3	0	0	0.3	1
A_L	0.5	0.5	0.5	0.5	0.5

Peer tutor selection
To test the peer tutor selection process we created five learners (see Table 2) and we assigned a set of test values to the parameter of the peer selection formula (cf. column 3, Table 1). Content competency as the most important element received weight 1. To simplify the preparation of the learners' data we set the weight of the Tutor Competency to 0. Furthermore, given that we only have five learners, we let them be always available, we assigned only one peer tutor per question, and we gave M, the bandwidth, value 1. Finally, we had learner 1 'ask' two of the 16 questions mentioned above. Next we assumed that a question is resolved by the learner with the highest rank and we asked the same questions once more to show the effect of workload. The results of this exercise on the behaviour of the model are given in Table 3.

Table 3. The selection results of question Q_5 and Q_{16}. The learner selected is in bold face.

	Question-id		Q5					
	Correlation	0.78	AN2: 'Using Internet Explorer'					
1	**Learner-id**			**L_2**				
	Tutor Suitability			0.83				
	WC	C_L	WE	E_L	WA	A_L	WT	T_L
	1	1	0.5	0.80	0.5	0.5	0	0
	Learner-id			L_3				
	Tutor Suitability			0.78				
	WC	C_L	WE	E_L	WA	A_L	WT	T_L
	1	1	0.5	0.6	0.5	0.5	0	0
2	**Learner-id**			**L_3**				
	Tutor Suitability			0.78				
	WC	C_L	WE	E_L	WA	A_L	WT	T_L
	1	1	0.5	0.6	0.5	0.5	0	0
	Learner-id			L_2				
	Tutor Suitability			0.76				
	WC	C_L	WE	E_L	WA	A_L	WT	T_L
	1	1	0.5	0.80	0.5	0.25	0	0

	Question-id		Q16					
	Correlation	0.77	AN11: 'Secure paying'					
3	**Learner-id**			**L_5**				
	Tutor Suitability			0.73				
	WC	C_L	WE	E_L	WA	A_L	WT	T_L
	1	1	0.5	0.40	0.5	0.5	0	0
	Learner-id			L_4				
	Tutor Suitability			0.38				
	WC	C_L	WE	E_L	WA	A_L	WT	T_L
	1	0.3	0.5	0.40	0.5	0.5	0	0
4	**Learner-id**			**L_5**				
	Tutor Suitability			0.73				
	WC	C_L	WE	E_L	WA	A_L	WT	T_L
	1	1	0.5	0.40	0.5	0.50	0	0
	Learner-id			L_4				
	Tutor Suitability			0.44				
	WC	C_L	WE	E_L	WA	A_L	WT	T_L
	1	0.3	0.5	0.40	0.5	0.75	0	0

Discussion of the results
The first results of the application of LSA suggest that it delivers as expected. The prototype identified the correct AN for 12 out of the 16 questions (75%). Moreover, two developers of the Learning Network in question, evaluated the text fragments, three for each question, that the prototype suggested. Ignoring the very small discrepancies in judgements between these experts, for about 6 to 7 of the questions, one or more text fragments were identified that in their opinion were useful for answering those questions. This figure seems far less accurate. The experts, however, indicated that 5 of the 16 questions posed were beyond the scope of the contents of the AN studied. As a consequence the AN could not possibly contain any useful fragments. Taking this into account, 6 to 7 questions with useful text fragments out of a total of 11 is a much better score (about 60%, for details, see Van Rosmalen, 2006) . Together the results are encouraging, taking into account the limited nature of the test. For about 75% of the questions the correct AN was identified; this means that in 75% of the cases content competent peer tutors may be selected. These will then be helped by providing them with text fragments; in the majority of the cases, at least one of those fragments was deemed useful by experts.

Also the first test of the selection rules is positive. The selections illustrate that we can balance the selection of peers with the help of workload and eligibility. In selection 1 the value of eligibility favoured Learner 2 over Learner 3, i.e., it prioritised the selection of a student in the same study phase. However, if we pose the question again the balance is shifted due to the workload of Learner 2. In selection 3 Learner 5 is selected based on his Content competency. But note that Learner 5 is selected again in selection 4. Learner 4 has not been involved yet, Learner 5 is simply too good. Obviously, the test has too limited a nature to allow one to draw general conclusions for the application of the selection rules in practice. How learners will behave and particularly how they will appreciate the selection rules should be assessed in empirical tests.

Conclusion

We started our discussion by arguing that a model is needed to organize and support learner related interactions in Learning Networks in a more efficient manner. For one type of support actions, answering content related question, we articulated our requirements and proposed a model. The test results of the first prototype showed that we were able to identify the relevant AN for some question, to select text fragments useful for answering the question, and to test our peer selection formula to the extent that it warrants carrying out an empirical study with 'real' students. This indicates that we can at least satisfy two of our requirements 'involvement' and 'support'. The first requirement '*the model has to alleviate the support task for the staff tutor while maintaining quality'* one can only test empirically. Most steps of the model are executed automatically. Nevertheless, empirical evidence has to shed light on how many questions will be resolved, what the quality of the answers is, and how much involvement of a staff tutor still is needed. The final requirement 'portability' is not yet met, but such is the nature of prototypes. The portability of the model is influenced by a number of factors. First of all, it should be possible to move the model from one system to another. This can be achieved by following for instance a service or an agent oriented approach. At a detailed level, the 'portfolio' of the learner should be accessible in an interoperable format. This can be achieved by applying the IMS-LIP standard (IMS-LIP, 2001). Moreover, for LSA to work efficiently, the course corpus has to be retrievable in a standard manner. This can be achieved by adopting the widely accepted IMS-CP standard (IMS-CP, 2003).

The next task now will be to carry out actual experiments. Questions to be addressed are (1) if and to which extent is the task of the staff tutor alleviated, (2) are peer learners capable and willing to answer questions, and (3) is there a measurable effect on the social cohesion of the Learning Network. Our first experiment, just started, will focus on question 1 and 2. Connected and subordinated to these questions, a number of critical conditions and parameters have to be determined, among others: the optimal size of the document corpus, the precise contents of the guidelines and the optimal size of the text fragments, the best size of the group, and the weights related to the selection of the peer tutors.

Acknowledgements

The authors would like to thank André de Jong and Maurice Brouwers, who contributed to the building of the test application. They are also indebted to an anonymous referee, who provided many comments that led to significant improvements of the manuscript. This project was funded by the OTEC Technology Development Programme in the project ASA (www.learningnetworks.org) and by the European Commission in TENCompetence (IST-2004-02787) (http://www.tencompetence.org).

References

Anderson T. (2004). Teaching in an Online Learning Context. In T. Anderson & F. Elloumi (Eds.) *Theory and Practice of Online Learning*, Athabasca University, 271-294. Retrieved September 1, 2004 from http://www.cde.athabascau.ca/online_book.

Bacsich, P. & Ash, C**.** (2000). Costing the Lifecycle of Networked Learning: Documenting the costs from conception to evaluation, *Association for Learning Technology Journal (ALT-J)*, 8 (1), 92-102.

Bartolic-Zlomislic, S. & Bates, A.W. (1999). Assessing the costs and benefits of telelearning: a case study from the University of British Columbia**.** *Network of Centers of Excellence (NCE)-Telelearning Project report*. Retrieved November 8, 2005 from http://research.cstudies.ubc.ca/nce/index.html.

Brouwers, M., Brouns, F., Van Rosmalen, P., Sloep, P. B., Kester, L. & Koper, R. (2006). ASA Tutor Locator. Retrieved April 10, 2006, from http://sourceforge.net/projects/asa-atl

Caron, J. (1999). *Design for the FAQ Organizer Application in Internet Dreams / CSCI 7000,* retrieved July 14, 2004 from http://my.unidata.ucar.edu/content/staff/caron/collab/InternetDreams.pdf.

Caron, J. (2000). *Applying LSA to Online Customer Support: A Trial Study*, retrieved April 18, 2004 from http://my.unidata.ucar.edu/content/staff/caron/faqo/faqoPaper1.pdf.

De Jong, A., Brouns, F., Van Rosmalen, P., Sloep, P., Kester, L., & Koper, R. (2006). GUP GTP Usability Prototype. Retrieved 10 April, 2006, from http://sourceforge.net/projects/gup

De Vries, F., Sloep, P., Kester, L., Van Rosmalen, P., Brouns, F., de Croock, M., Pannekeet, C. & Koper, R. (2005). Identification of critical time-consuming student support activities that can be alleviated by technologies. *Research in Learning Technology (ALT-J),* 13 (3), 219-229.

Dumais, S. (2003). Data-driven approaches to information access. *Cognitive Science* 27 (3), 491-524.

Ellis, A.L., Longmire, W.R. & Wagner, E.D. (1999). Managing Web-Based Training. *American Society for Training & Development.*

Fantuzzo, J. W., Riggio, R. E., Connelly, S., & Dimeff, L. A. (1989). Effects of reciprocal peer tutoring on academic achievement and psychological adjustment: A component analysis. *Journal of Educational Psychology.* 81, 173-177.

Foltz, P.W., Laham, D. & Landauer, T. (1999). Automated essay scoring: applications to educational technology. *World Conference on Educational Multimedia, Hypermedia and Telecommunications*, Vol. 1999, 1, 939-944.

Foner, L.N. (1997). Yenta: a multi-agent, referral-based matchmaking system. In W. Lewis Johnson (Ed.), *Proceedings of the first international conference on Autonomous agents***,** Marina del Rey, California, United States. Retrieved October 13, 2004 from http://doi.acm.org/10.1145/267658.267732.

Gaston J. (2003). Interactive Agents: responding to student queries “naturally”. Presentation at *EDUCAUSE 2003*, Los Angeles. Retrieved September 10, 2004 from http://mysiteagent.com/educause.

Giles, J. T., Wo, L., & Berry, M. W. (2001). *GTP (General Text Parser) Software for Text Mining*. Retrieved January 10, 2005 at: http://www.cs.utk.edu/~berry/papers02/GTPchap.pdf

Graesser, A., Wiemer-Hastings, P., Wiemer-Hastings, K., Harter, D., Person, N., & the Tutoring Research Group. (2000). Using Latent Semantic Analysis to evaluate the contributions of students in AutoTutor. *Interactive Learning Environments*, 8,149-169.

Guzdial, M. (1997). Information Ecology of Collaborations in Educational Settings: Influence of Tool. In *Proceedings of the CSCL’97*, Toronto.

Howell, K. (2003). *Question Generation and Answering Systems R&D for Technology-Enabled Learning Systems Research Roadmap*, retrieved November 8, 2004 from http://www.thelearningfederation.org/qa.pdf.

IMS-CP (2003). *IMS Content Packaging. Information Model, Best Practice and Implementation Guide*. Version 1.1.3 Final Specification IMS Global Learning Consortium Inc. Retrieved May 26, 2004 from http://www.imsglobal.org/content/packaging/.

IMS-LIP (2001). *IMS Learner Information Package, Information Model, Best Practice and Implementation Guide*. Version 1.0 Final Specification IMS Global Learning Consortium Inc. Retrieved January 22, 2004 from http://www.imsglobal.org/profiles/.

Janssen, J., Tattersall, C., Waterink, W.,Van den Berg, B., Van Es, R., Bolman, C., & Koper, R. (in press). Self-organising navigational support in lifelong learning: how predecessors can lead the way. *Computers & Education*.

Kester, L., Sloep, P. B., Van Rosmalen, P., Brouns, F., Koné, M., Koper, R. (in press). Facilitating Community Building in Learning Networks Through Peer-Tutoring in Ad Hoc Transient Communities. *International Journal on Web Based Communities*.

Koper, E. J. R. (2004). Use of the Semantic Web to Solve Some Basic Problems in Education: Increase Flexible, Distributed Lifelong Learning, Decrease Teacher's Workload. Journal of Interactive Media in Education, 2004 (6). Special Issue on the Educational Semantic Web. ISSN:1365-893X. Retrieved September 28, 2004 from www-jime.open.ac.uk/2004/6.

Koper, R. (2006). The TENCompetence Domain Model. *TENCompetence Project Report*. Retrieved July 10, 2006 at: http://dspace.ou.nl/handle/1820/649

Koper, E. J. R., Giesbers, B., Van Rosmalen, P., Sloep, P., Van Bruggen, J., Tattersall, C., Vogten, H. & Brouns, F. (2005). A design model for lifelong learning networks. *Interactive Learning Environments* (1-2), 71-92.

Landauer, T, Foltz, P.W. & Laham, D. (1998). An introduction to latent semantic analysis. *Discourse Processes* 25, 259-284.

Person, N.K., Gholson, B., Craig, S.D., Hu, X., Stewart, C.O., & Graesser, A.C. (2001). HURAA: An interactive web-based agent that optimizes information retrieval in a multi-media environment. In C. Montgomerie & J. Viteli (Eds.), *Proceedings for ED-MEDIA 2001: World conference on educational multimedia, hypermedia and telecommunications* (1476-1481). Norfolk, VA: Association for the Advancement of Computing in Education.

Salmon, G. (2000). *E-moderating: the key to teaching and learning online*. London: Kogan Page.

Van Bruggen, J., Sloep, P., Van Rosmalen, P., Brouns, F., Vogten, H., Koper, R., Tattersall, C. (2004). Latent semantic analysis as a tool for learner positioning in learning networks for lifelong learning. In *British Journal of Educational Technology*, 35 (6), 729 – 738.

Van Rosmalen, P., Sloep, P., Brouns, F., Kester, L., Koné, M. & Koper, R. (2006). Knowledge matchmaking in Learning Networks: Alleviating the tutor load by mutually connecting learning network users. *British Journal of Educational Technology,* 37, (6), 881-895.

Vassileva, J., Detters, R., Geer, J., Maccalla, G., Bull, S. & Kettel, L. (2001). Lessons from Deploying I-Help. Workshop - Multi-Agent Architectures for Distributed Learning Environments. *Proceedings of International Conference on AI and Education*, San Antonio, Texas, May, 2001, 3-11.

Vivacqua, A. & Lieberman, H. (2000). Agents to assist in finding help. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 65–72). ACM Press New York, NY, USA. The Hague, The Netherlands. Retrieved October 13, 2004 from http://doi.acm.org/10.1145/332040.332408.

Wiley D. (2004). *Scalability and Sociability in Online Learning Environments,* retrieved September 1, 2004 from http://wiley.ed.usu.edu/docs.

Woudstra, A., Huber, C. & Michalczuk, K. (2004). Call Centers in Distance Education. In T. Anderson & F. Elloumi (Eds.) *Theory and Practice of Online Learning*, Athabasca University, 295-317. Retrieved September 1, 2004 from www.cde.athabascau.ca/online_book.