

Problem-based Web-based Teaching in a Computational Linguistics Curriculum

Kai-Uwe Carstensen / Michael Hess (Zurich)

Abstract

This paper presents an approach for combining web-based learning (WBL) with problem-based learning (PBL), aiming at computer aided learning (CAL) support for introductory lectures to Computational Linguistics. Contrary to most current learning paradigms, we neither follow the "platform + content"-approach of current learning management systems used for distance learning, nor focus on the collaborative aspects of PBL. Instead we propose a *text-centred* concept for *individual* learning featuring *problem-based* interactive learning applications (TIP).

1 Introduction

There is no doubt that learning in groups is in general more fun, often more stimulating, and sometimes more successful than learning individually. This simple insight is reflected in the computer supported collaborative work (CSCW) tools omnipresent in class-management systems (e.g., WEBCT, cf. www.webct.com) and web-based learning (WBL) platforms (e.g. MILCA, milca.sfs.uni-tuebingen.de). It is also corroborated by the results of the new problem-based learning (PBL) paradigm which is essentially a collaborative work learning strategy used together with a new concept of teaching (Rhem 1998).

Not always is group learning reasonable or possible, however. While preparing for exams most students will have to work through course materials individually (in addition to group work with fellow students), as their personal level of competence will vary widely. In some teaching contexts, extensive tutoring of student groups would be desirable, but due to financial restrictions this may not always be possible. Web-based tutorial systems with up-to-date progress monitoring features provide the student with individual and adaptive help and feedback. And yet, in assigning the student a passive role in learning, they are in conflict with one of PBL's most basic demands: active learning as a prerequisite for life-long learning. In addition to that, both WBL and PBL cannot easily be conceived of as natural extensions of continually evolving lectures and their corresponding lecture notes, which requires a text-centred approach (as opposed to an environment-based one) to learning. It would therefore be beneficial to find a synthesis of these antagonistic approaches within the computer aided learning (CAL) paradigm, and to integrate traditional lecture notes with WBL and PBL.

In the years 2000-2002, the University of Zurich funded well over 70 projects dedicated to the use of information and communication technology (ICT) in implementing web-based learning and tutoring (Seiler-Schiedt 2003). In this framework, the Institute of Computational Linguistics realized a project on web-based tutoring for introductory lectures in Computational Linguistics. The goal of this project was to develop a CAL concept that a) smoothly integrates with and extends the lecture notes of existing lectures (so-called "blended learning"), b) is oriented towards self-paced learning by the individual student, and c) realizes core ideas of problem-based learning. We assumed that instances of this concept could be easily combined with CSCW-tools (email, chat, whiteboard etc.) and integrated into an encompassing class-teaching environment (which also includes assessment tools like quizzes, tests, etc.). Therefore, we concentrated on the content- and task-related aspects leading to a synthesis of PBL and CAL. In the following we will first elaborate on PBL in some more detail and will then describe our "TIP approach", i.e. *Text-centred, Individual-oriented, Problem-based* CAL.

2 Problem-based-learning

The past decades have seen exciting developments in educational and technological approaches to learning. From the educational viewpoint, classical teacher-student instructivism is increasingly replaced by *constructivism*, a philosophical view on how we come to understand or know. Constructivism assumes that "knowledge" is not a pre-existing entity to be assimilated by the learner but must rather be "constructed" by the learner based on previous knowledge and overall views of the world. According to Savery/Duffy (1995), constructivism can be characterized by three primary propositions.

First, knowledge is *in* our interactions with the environment. Understanding is not only a function of the content, but also of the context and of the activity and goals of the learner.

Second, the stimulus for learning is a cognitive conflict or puzzlement for the learner. This determines the organization and nature of what is learned.

Third, understanding is influenced by the processes associated with the social negotiation of meaning. What we call knowledge is not fixed forever but usually open to discussion or scientific refutation. Quite opposite to what happens in school, we therefore constantly look for alternative views and additional information in our social environment in order to judge the viability of our understandings.

The first proposition corresponds to the basic tenet of *situated learning* (Clancey 1995) that an *authentic context* is important for learning. Traditional instructivist learning is often quite distinct from authentic activity, and many of the activities undertaken by students are unrelated to those performed by practitioners in their everyday work. In the model of *cognitive apprenticeships* (Brown et al. 1989), authenticity (and with it, meaningful learning) is achieved if the learner can act as an apprentice observing the 'community of practice'.

Being rather an instructional strategy than a theory of learning, *problem-based learning* (PBL) shares the basic assumptions of constructivism but puts an emphasis on the second proposition: "The principal idea behind PBL is that the starting point for learning should be a

problem, a query or a puzzle that the learner wishes to solve" (Boud 1985). "Cases", i.e. real-life scenarios that exemplify the problem to be solved, provide the authentic context for learning, and students try to achieve a solution to the problem in a *collaborative manner*, which includes dealing with *alternative viewpoints* on the problem (see figure 1).

PBL presents a radical departure from instructivist learning strategies in the sense that puzzlement (deprecated in traditional learning styles) is taken literally: Problems are typically ill-structured so that more information is needed to gain an understanding of the problem than is immediately available. This *understanding* becomes the center of interest in PBL, as opposed to the focus on *achieving solutions* in other learning styles.

Another point of departure lies in the roles of teacher and student (which are relatively stable in traditional teacher-centred learning). In PBL, learning is *student-centred* in that the students (have to) have the overall responsibility for the learning process and take different roles, while the teacher recedes into the background and acts only as a facilitator/guide/coach/tutor (and as a scaffolding of the learning process) or even as a co-learner. Thus, PBL promotes *active, self-directed* learning and supports acquiring skills needed for life-long learning. The idea of PBL is nicely summarized by Duch:

Problem-based learning (PBL) is an instructional method that challenges students to "learn to learn," working cooperatively in groups to seek solutions to real world problems. These problems are used to engage students' curiosity and initiate learning the subject matter. PBL prepares students to think critically and analytically, and to find and use appropriate learning resources (Barbara Duch, www.udel.edu/pbl/, last visited: Oct. 2003).

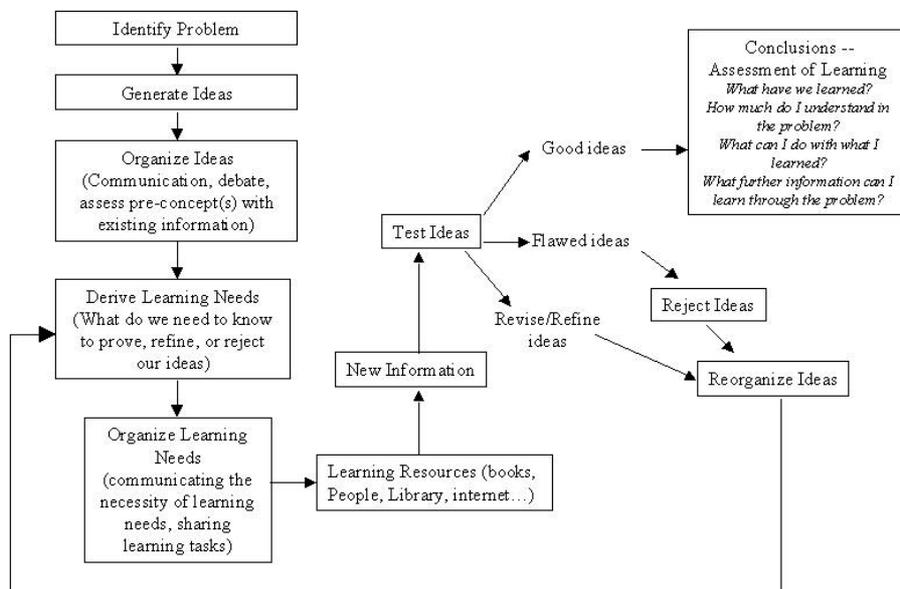


Figure 1: Problem-based learning process¹

¹ Original appeared in Wang, Thompson, Shuler & Harvey (1999), *Problem-Based Learning for Science Teacher's Professional Development*. Paper presented at the 1999 AETS Annual Conference, Austin, Texas.

3 Computer-aided learning

The terminology used in educational technology is diverse and often confusing. For the purpose of this paper, we therefore simply define CAL as the most general term for the use of information and communication technology (ICT) in learning. Broberg (2000: 29ff.) distinguishes three generations of CAL-environments. *First-generation systems* basically automate the information presentation and assessment cycle with a computer, sticking to the instructivist tradition. Examples are computer aided instruction (CAI), computer based training (CBT) and computer based learning (CBL) systems. *Second-generation systems*, for which Intelligent Tutoring Systems (ITS) or current adaptive (educational) hypermedia systems (Brusilovsky 2001) are representative, add complexity in presentation, modelling, and analysis. They allow multiple paths through the curriculum, present the material in different ways, and analyse the errors that the learners make. Besides modelling the domain (the expert model) and the teaching strategies (the tutor), they also model the learner's state of knowledge and react accordingly. However, "[t]he reason for the failure of many ITS was their ambitious approach. They were designed top-down and were meant to be comprehensive. They tried to deal with the *why* aspect of knowledge in absence of its natural channel of communication i.e. language and linguistic representation" (Kinshuk/Patel 1996: 225). Because of this, *third-generation systems* usually focus on the use of *intelligent tutoring tools* (ITT) supporting learning and tutoring in Computer Integrated Learning Environments (CILE, cf. Patel/ Kinshuk 1997) or Web-Based Interactive Learning Environment (WBILE, cf. Puntambekar 1999). Among the purposes of CILE, Patel/Kinshuk (1997) mention the following:

- to employ useful software tools within the overall learning environment consisting of human teachers and educational technologies;
- to add intelligence to the software tools to provide a degree of support to students, enabling them to work by themselves;
- to let intelligent tutoring evolve from practically useful applications, in a bottom up fashion (through vertical and horizontal integration) rather than be designed top-down.

The more general notion of *cognitive tools* has its pedagogic base in the constructivist ideas of knowledge and learning. Broberg (2000: 17) lists three dimensions of cognitive tools that span a space of their possible applications in CILEs:

- *Control*: "concerning where the control over the learning situation and the artefact is, ranging from total teacher control to total learner control."
- *Generativity*: "concerning the view of learning and knowledge, permeating the learning situation, ranging from pure presentation to genuine creation."
- *Engagement*: "concerning the way the learners act in the learning situation, ranging from passive to active."

As can easily be seen, this broad conception of educational technology is quite compatible with PBL approaches. The following section reviews how PBL can be integrated into CAL (which nowadays can probably be identified with web-based learning) and vice versa.

4 From pure to impure PBL: PBL and ICT

As Camp notes, "PBL is an innovation which has definitely "caught on" in medical schools and in numerous other settings" (Camp 1996: 6). Especially in some medical departments, e.g. at the universities of McMaster (Canada), Maastricht (Netherlands), and Berne (Switzerland), it has led to a curricular reorganisation towards small-group, tutor-led, student-centred modes of teaching and learning. Reflecting on the core characteristics of PBL, Camp therefore proposes that "any program which does not place students in tutorial groups of, say, 5-10 students is not "pure" PBL" (Camp 1996: 5). There is no unanimity regarding the role of small-group learning for PBL, however, in view of the vast number of different applications of PBL in a growing number of disciplines. Barrows (Barrows, 1986) already proposed six variants of PBL spread along a continuum from lecture-based cases to more completely problem-based approaches. Others rather put the combination of problem-basedness, learner-centredness, and situatedness as core characteristics of PBL to the fore (Charlin et al. 1998).

Although the benefits of pure PBL must be acknowledged ("PBL *does* provide a more challenging, motivating and enjoyable approach to education", Norman/Schmidt 2000: 727), it is not without problems: "We believe that PBL *has* been oversold by its advocates, promising enormous benefits and largely ignoring the associated resource costs" (Norman/Schmidt 2000: 721). In that respect, Farnsworth (1994) mentions the following three points as major objections to PBL: (1) PBL is an inefficient method of instruction since it requires students to gather information through self-directed learning, (2) PBL is perceived as costly since it requires a greater investment of faculty time to function as tutors, and (3) PBL is more difficult and costly in terms of evaluation of student learning (Farnsworth, 1994, p. 137 cited in Hoffman/Ritchie 1997: 102).

Considering the teaching of a Computational Linguistics introductory course, these objections can be made clearer: It is quite unreasonable to assume that complex theories, methods, and implementational issues of the discipline can be learned in the pure PBL fashion in reasonable time. It is practically impossible that a pure PBL curricular version *replaces* the traditional lecture-and-practice version. In fact, being able to teach growing numbers of students without hiring more staff was an important motivation for the funding of ICT projects at the University of Zurich. It is therefore desirable to have an individualized "impure" version of PBL provided that facilitation and scaffolding are not negatively affected. It should furthermore be relatively easy within an ICT discipline to integrate assessment into an appropriately conceived PBL-CAL scheme (instead of doing PBL-style evaluation).

Recent CAL-oriented approaches seek to overcome these problems by using (interactive) multimedia (Albion 2000b, Hoffman/Ritchie 1997, Koch/Teege 1999). Hoffman/Ritchie (1997: 102ff.) list the following advantages of integrating multimedia in PBL:

- *Fidelity*: The fact that multimedia (MM) makes it possible to present problem descriptions that resemble "real life" situations more closely (i.e., not only verbal descriptions).

- *Representational richness*: The ability to provide a robust representation of the problem environment thus improving the learner's comprehension of the situation and the action in it.
- *Time and timeliness*: The capability of MM to instantly provide information (and expert knowledge) when needed, and to foster the manipulation of time-based phenomena.
- *Individualization*: The possibility to adapt to the individual learner's background, interests, competence etc.
- *Assessment*: The possibility to monitor and evaluate the learner's behaviour while interacting with a simulated environment.
- *Efficiency*: The possibility of alleviating time problems in the implementation of PBL, both for instructors and learners. This may include adaptive scaffolding.
- *Increased power of agency*: The ability to break out of the real-life limitation through non-risk try-out simulations

Most CAL-oriented approaches try to support the collaborative aspect of PBL (Koch/Teege 1999, Miao 2000). Koch/Teege describe an environment for multimedia-rich problem-based learning in computer science that comprises the following components: the electronically available *lecture notes* of the group, a *multimedia presentation* which contains the information necessary to solve a case study, a number of *tools* which assist the students to analyse and model case studies, and a *working environment* which provides a generic framework. Even pure PBL has been implemented in computer science introductory courses (Cavedon et al. 1997, Barg et al. 2000). However, few approaches exist that implement an individualized version integrating PBL into CAL (Albion/Gibson 1998, Albion 2000a). Since it has been argued (Dale et al. 2003) that CL should play - in general - a more prominent and at the same time more practice-oriented role in undergraduate computer science curricula (due to the growing importance of language technology in ICT), we regard the TIP approach as an important contribution in that respect.

5 Teaching a CL introductory course: The TIP approach

5.1 Motivating TIP

In most fields, an important aspect of becoming an expert is the ability to *bridge the gap* between the theoretical and practical side of the corresponding discipline, and it is a characteristic feature of PBL that precisely this ability is trained. There are some points in which Computational Linguistics differs substantially from medicine (as the prototypical PBL discipline), however (see table 1).

	Medical domain	CL
Number of relevant problems/cases	Large	Small
Basis for problem solving	Facts, rules	Structural considerations
Solution of problem	Structurally simple	Structurally complex

Table 1: Differences between the medical domain and CL

As Table 1 shows, CL does not have the characteristics that have led to (the success of) PBL in the first place: a vast number of cases/problems, an ever-growing number of facts drowning the student, and the fact that problem solutions (i.e., diagnoses) are structurally simple (contrary to the complexity of the problem solving process).

In CL, there is only a small number of problem types (e.g., summarizing a text, generating an object description based on database information, retrieving information from texts in reply to a query phrased in natural language), but the corresponding problem solutions (systems, modules, algorithms etc.) are structurally complex and require structural considerations (on the levels of theory, method, and implementation) during problem solving.

Because of that, the learner's understanding in CL might rather be hindered by negotiating with other learners' opinions during a pure PBL process, although *individual* understanding is even more important in this discipline.² This confirms the above critique of pure PBL as an inefficient method of instruction for CL. Furthermore it is evident that pure PBL would be much too costly in terms of time, number of faculty, and assessment effort.

Apart from these aspects, however, benefits can be expected from problem-based, student-centred, active, self-directed, situated learning in a framework of web-based tutoring (PBL-CAL). We therefore propose to synthesize instructivist aspects of presenting complex content with impure PBL aspects of *anchoring* the acquired knowledge in problem-based application scenarios, each addressed to the individual learner. In addition to that we propose to closely connect lecture notes and problem-based parts, which means that links to the latter are properly placed in their theoretical context within the former. From this does result what we call the TIP (*text-centred concept for individual learning featuring problem-based interactive learning applications*) approach to PBL-CAL.

5.2 Aspects of TIP

TIP consists of the following aspects:

Text-centredness. At the heart of TIP are the lecture notes delivered as a hypertext document in PDF. This is *not* a collection of slides but a properly worded and hierarchically structured form of 'teaching as telling'. The latter is taken literally, because we assume that the coherence and hence, the readability, of the text is essential for an understanding of the content. As a hypertext and skeleton of TIP, this main document contains various links to further information sources:

² It should be mentioned here that reasonable collaborative work (e.g., in the form of student projects) is usually integrated in informatics curricula.

- Subdocuments *providing more detailed information* on some topic (i.e., on a higher level of granularity). These documents are part of the main document and are structured according to the didactic principles of the latter.
- Separate documents covering a specific topic by *providing additional background material* (like book excerpts, web pages, newlist-entries). These documents are (didactically) *unstructured*.
- *Learning units*, that is, separate documents that *provide a didactically well-structured treatment* of a subtopic.
- *Interactive Learning Applications (ILAPs)*: program demos, automatized practices and instances of an interactive learning environment. *Problem-based ILAPs (PILAPs)* mark the "P" of TIP.
- Various kinds of tests as part of the self-assessment.
- Global resources (lexica, corpora, programs, glossary etc.), which exist independently of the lecture.

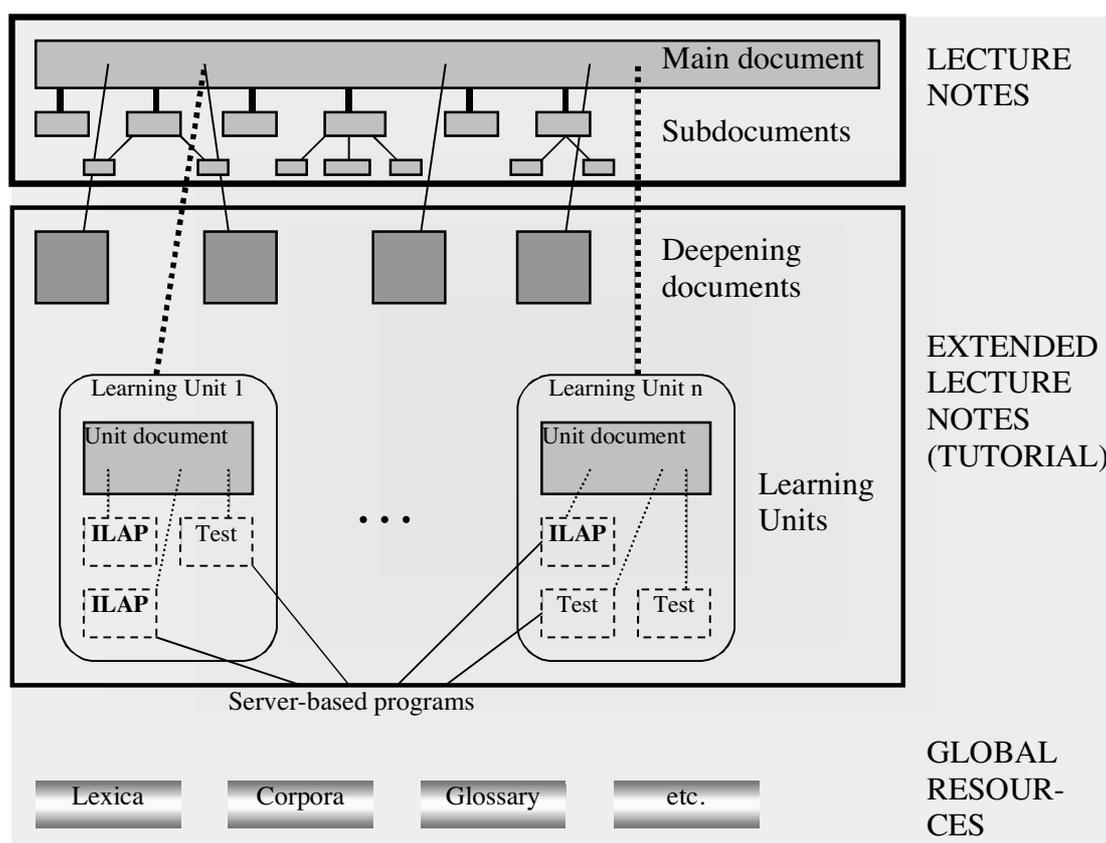


Figure 2: Text-centredness in TIP

An advantage of text-centredness, which is shown in figure 2, is the smooth transition from lecture (notes) to problem-based learning with ILAPs, as the latter is structurally integrated in the former. This helps to conceptually bridge the above-mentioned gap between the theoretical and practical sides of CL. We also know that students prefer to print out (parts of) the lecture notes. It is therefore natural to work with the same document on the computer

(linking to learning units with problem-based ILAPs), as opposed to the conceptual distance of lecture (notes) and platform-based ITS tutoring like ELM-ART (Weber/Specht 1997) or ILIAS (<http://www.ilias.uni-koeln.de>). While we lose the "intelligence" of ITS systems, we expect a corresponding gain with the PILAPs as smaller-scale ITT.

Individualization. Contrary to the mainstream ideas that favour group learning, we address the individual learner and defocus the communicative aspect of PBL. This is due to the fact that establishing a pure PBL framework must be regarded as unrealistic (wrt. to costs involved) and unreasonable (wrt. the complexities of the content). Because of the close connection between lecture and tutoring (as opposed to distance learning), we do not offer CSCW tools (although it should be quite easy to integrate TIP into a web-based courseware that inherently provides these facilities). However, we encourage (but not force) our students to work collaboratively with the PILAPs as PBL tools.

Impure PBL. As an ICT discipline, Computational Linguistics offers the rare opportunity for the authors of web-based materials to create authentic contexts for learners: realistic conditions (computer setting, textual materials etc.) can be set up straightforwardly, and the learner can -even without much multimedia effort- be quite easily given the task of solving ill-structured real-life problems with given software tools and resources. Of course, the quality and success of this impure PBL procedure crucially depends on the interactive learning environment presenting the problem and providing the means for problem manipulation, access to relevant materials, and scaffolding in an open-ended learning scenario. Below we will explain the problem-based interactive learning environment (PILE) we have developed, and will give an example of its use.

6 PILE: A problem-based interactive learning environment

The PILE is basically a web-page containing a java-applet that acts as an interface to server-side programs. While these programs are the *software tools* of the domain of Computational Linguistics (either generally available or developed especially for a PILAP), the java-applets represent the *cognitive tools* driving problem-based learning in the PILE. In developing the PILE we gave first priority to the following design aspects:

- *Simplicity*: Complexity should only arise from content and problem; virtually no effort should arise from interacting with the PILE.
- *Overview*: Although there can be a lot of subcomponents in a PILAP, all of them should be visible on one screen. Scrolling of the PILE web page should be unnecessary.
- *Magnification*: Text fields may become quite small (because of the overview criterion). Therefore, each relevant text field has a magnify button, which opens a separate, resizable window with the field content.

A PILE window generally consists of three components: The *info-bar* which gives access to all information relevant for using the PILE and for understanding a specific PILAP; the *resource-field* which gives access to all resources relevant for using the PILAP; the *work-area*

which is the main locus of interaction of the learner with the PILAP. Figure 3 shows the PILAP "Information filtering with regular expressions" as an instance of the PILE.

The info-bar consists of the following links (each of which opens a window with pertinent information when clicked on):

- *Information about the PILAP* ("Info"): This describes the current PILAP and presents the problem to be solved and the case to deal with (see figure 4).
- *Task* ("Aufgabe"): This names a specific task of the learner given the problem.
- *Hints* ("Tipps"): Here the learner can find hints when stuck in the process of problem solving.
- *Help regarding the current PILAP* ("Hilfe"): This gives detailed information about specific properties of the current PILAP (functionality of interface items, sequence of operations to perform, descriptions of resources).
- *General help concerning the PILE* ("Allgemeine Hilfe"): This gives a general description of the functionality of the (parts of the) PILE.

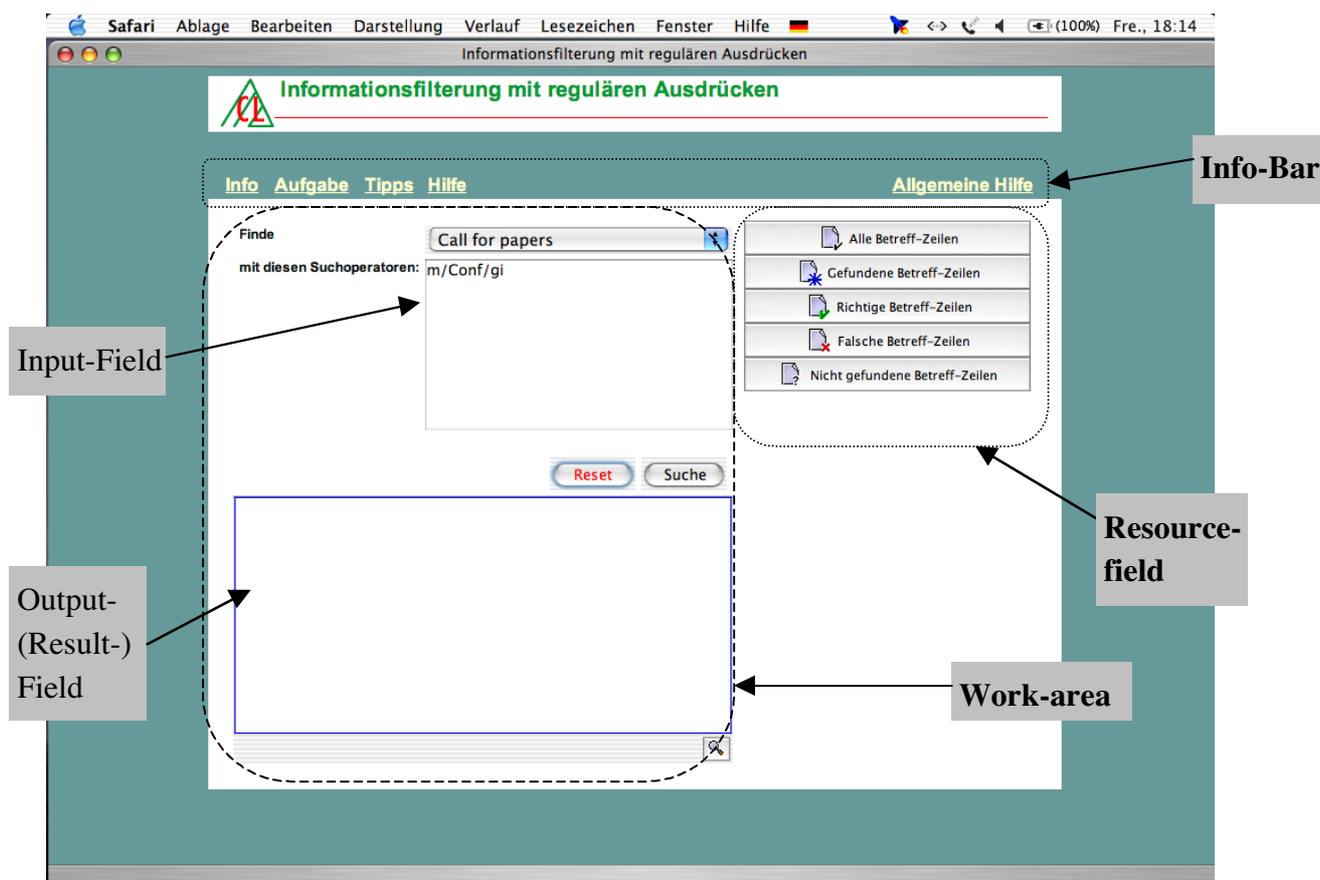


Figure 3: The structure of the PILE

The work area contains the input-, output-, choice-, decision-, and action-related parts of the PILE. In this area, most of the interactivity takes place. The resource field contains a set of buttons each of which opens a (mostly editable) window with relevant textual material (which could be generalized to multimedia information).



Figure 4: Information window

7 PILAPs: Problem-based Interactive Learning Applications

7.1 An example: Information filtering

PILAPs are instances of the PILE that are accessed via hyperlinks from somewhere in the lecture notes or learning units (that is, from within the *problem context* of the PILAP). For example, *filtering of information* in texts is described as one of the field's applications in the main text of the lecture notes of "Introduction to computational linguistics". At the end of the corresponding section, there is a link to a learning unit about "regular expressions" which are widely used in language technology for shallow text processing.³ Here the student is first given a detailed account of how arbitrary pieces of text can be theoretically specified in a regular language format, and practically selected/extracted with corresponding operators.⁴ After that, applications of regular expressions within language technology are presented in some detail. Finally, there is a short description about an ILAP in which the student is given a list of newstlist headers ("Betreff-Zeilen") from which only headers of a given type (e.g., call for papers, spam) are to be selected. The student is then linked to the *information filtering* (IF) PILAP shown in figure 3, and is immediately presented detailed information in the form depicted in figure 4. While this shows how PILAPs are embedded in their instructivist textual context, we will now discuss why PILAPs support problem-based learning.

³ where "shallow processing" is characterized by the use of simple methods and efficient algorithms.

⁴ This can be immediately practiced with simple ILAPs.

7.2 PBL aspects of PILAPs

Albion (2000b) lists nine principles of PBL and describes how they are realized in his problem-based interactive multimedia framework. In the following, we will do the same wrt. the TIP approach by using the IF-PILAP as example.

Begin with an authentic problem. Every PILAP begins with the description of a problem (in fact, the corresponding information window pops up when the PILAP is started). Although somewhat "scaled down" for the student of an introductory course, PILAP-problems correspond to those faced by experts in the field. We follow PBL principles, of course, in not expecting a perfect or even complete solution, and allowing open-ended learning. We even regard the awareness and understanding of the problem on the way to its solution as an important part of problem-based, situated learning (for the CL introduction). In the IF-PILAP, it is the problem of how to get satisfactory information filtered from some given data by using regular expressions. In this case, getting a "feeling" for why this is not possible is the expected result of the PILAP (which goes beyond the expected problem solution).⁵

Incorporate relevant cases. We use real-life cases in our PILAPs to foster example-based learning. In the IF-PILAP, information filtering applied to real-life text data (2000 lines of newlist headers) represents the case.

Represent multiple viewpoints. We do not use multimedia or communication for this aspect. Instead we distinguish between the problem and various related (sub)tasks. Working on different tasks within one problem domain then allows viewing the problem from different angles. In the IF-PILAP, the learner has to choose between, e.g., extracting call for paper announcements or extracting/filtering spam.⁶

Stimulate activation and elaboration of knowledge. This point is inherent in achieving progress within a PILAP. The learner *must* activate and connect his passive knowledge of the domain (e.g. regular expression syntax) and reflect upon the structure of the data in order to be able to input proper search expressions.

Scaffold learner performance. General support is given by the learning unit or that part of the lecture notes a PILAP is embedded in (which is an important aspect of the overall TIP approach). As instances of a tutorial environment, PILAPs offer a variety of additional support. Learners can easily access the info bar for help regarding the PILE, the PILAP, or the solution of the problem (via the hints). The design and structure of the PILAP itself is meant to guide (and not to mislead or confuse) the learner in his interaction (e.g., the inactivity of an action button if some input is missing). The results produced by the PILAP can be regarded as an immediate feedback. Sometimes, aspects of the results can be inspected via the resource

⁵ In fact, besides introducing techniques of shallow natural language processing, part of the didactic functionality of our basic PILAPs is to show what *cannot* be done with standard methods of computer science (like regular expression search). Being able to experience these limits as opposed to simply being told adds another value to the TIP approach.

⁶ Note that PILAPs provide what Barab et al. (2000) call *factual, procedural, and task* authenticity, which correspond to their real-life data, use of methods, and tasks, respectively.

field (in the IF-PILAP, these are for instance "missed news headers", "news headers found", "news headers to be found" for the actual task). Furthermore, there is (sporadic) context sensitive feedback guiding the learner (e.g., in our PILAP for date expression recognition, examples for still unrecognised expressions are presented in a pop-up window).

Provide a strong narrative line. As we do not make heavy use of multimedia, this is a minor point in the TIP approach. Depending on the complexity of a PILAP, however, the work area can be cluttered with subcomponents (selection boxes, input fields, action buttons etc.), being a possible source of confusion for the learner. In order to induce structure we therefore use a corresponding top-down order as iconic representation of the ideal sequence of user actions.

Provide access to relevant information. We provide a resource field with links to relevant material most of which opens in resizable, often editable, windows.

Encourage self-evaluation. In some PILAPs, the data are, unknown to the user, annotated in order to allow evaluation of the learner's results. Aspects of the evaluation are then presented to the learner as a feedback simulating an experts response to the learner's action. In the IF-PILAP, search results for the regular expressions devised by the learner are commented wrt. precision and recall and some other aspects of the result (see figure 5). On the basis of these comments, learners can evaluate their understanding of the problem and the quality of the proposed solution.

Support individual and collaborative learning. As has been explained before, only individual learning is supported in the TIP approach.

While the PILE fulfils all of the above-mentioned criteria for computer integrated learning environments, the PILAPS can be rightly regarded as intelligent tutoring tools that make use of software tools existing in the CL domain. As cognitive tools, PILAPs maximize the values for control, generativity, and engagement, which is in full accord with PBL. Some aspects of PILAPs deserve to be mentioned in particular:

- *Active learning.* The student *must* be active or else virtually nothing happens.
- *Fault-tolerance.* Wrong actions of the student are regarded as part of learning and can be made infinitely often. Students have to monitor the effects of their actions and have to re-act accordingly. For this reason, PILAPs can be regarded as "experimental" learning environments.
- *Feedback* is immediate in a very basic sense: Students can directly see what is caused by their actions ("what-you-see-is-what-you've-done").
- *Motivation* is not externally given but is student-generated (e.g., self-reward through progress made).
- *Self-directedness.* The learner is fully responsible for his learning progress.
- *Student-centredness.* There is no teacher in PILAPs. All the tutoring is in their design and in the info-bar.

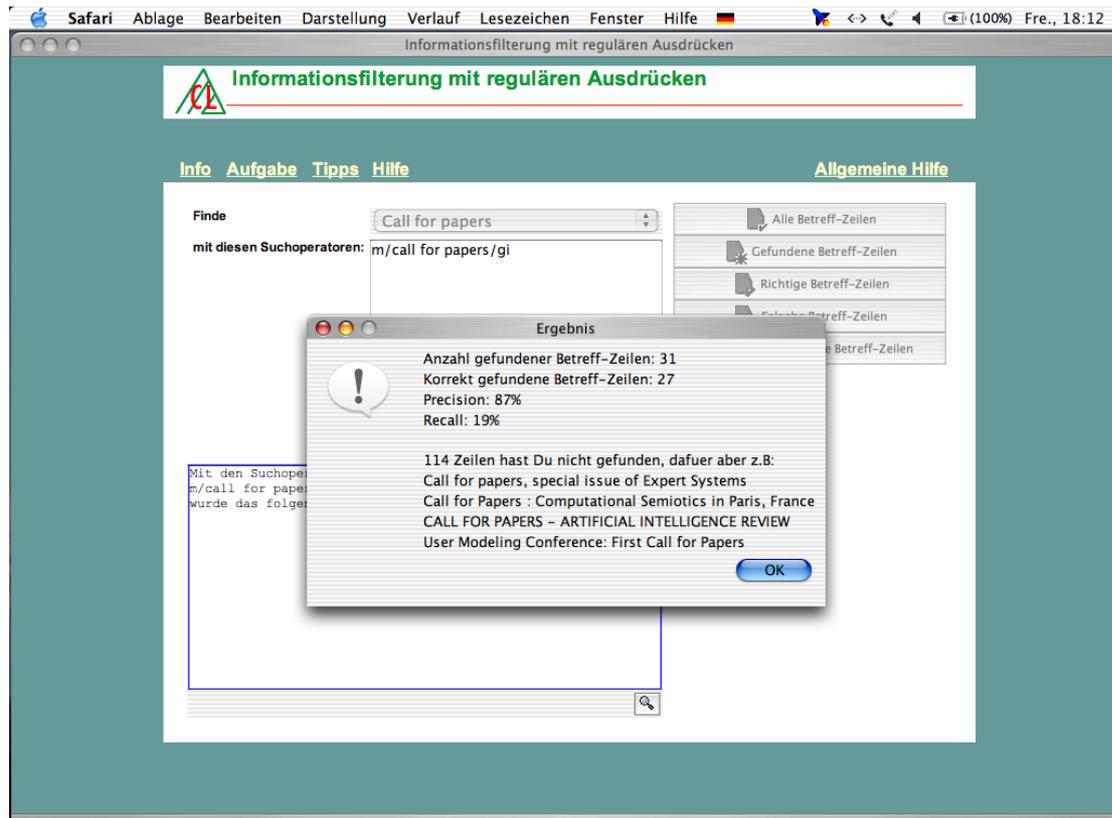


Figure 5: Result presentation in the IF-PILAP

Although multimedia is not employed in PILAPs, most of the advantages listed above for the use of multimedia also apply to PILAPs. We must admit that this is due to the privileged situation of Computational Linguistics as regards the large overlap of learning scenario and real life. However, the resource field of a PILAP fulfils the same function as the multimedia presentation in the Koch/Teege (1999) approach, and could be easily adapted to multimedia requirements.

8 Summary

We have presented an approach to web-based learning that combines aspects of problem-based learning and computer-aided learning. We have shown that a synthesis of traditional instructivist lecture and complementary problem-based constructivist learning ("blended learning") can be achieved with a text-based teaching concept in which extended lecture notes provide a suitable context for embedded (links to) intelligent tutoring tools in the form of problem-based interactive learning applications (PILAPs). With TIP, we address the individual learner. Although this runs counter to PBL's core feature, group learning, we have argued -in line with other criticism of the general feasibility of pure PBL- that an individualized PBL-CAL implemented along the lines of the TIP approach is a promising new form of web-based learning, as it comprises all of PBL's other advantageous features (student-centred, active, self-directed learning) while using ICT for teaching. In winter semester 2003/2004, TIP has been introduced into teaching the introductory lectures of Computational Linguistics at the University of Zurich (see www.cl.unizh.ch and, in particular, www.ifi.unizh.ch/cl/study/webcl.html).

Acknowledgements

We are grateful to Cerstin Mahlow for many fruitful discussions during the WebCL project, and to Jennya Dobрева for implementing the PILE. Our thanks also go to the two anonymous reviewers of this paper for their helpful comments.

References

- Albion, Peter R./Gibson, Ian W. (1998): "Designing multimedia materials using a problem-based learning design." In: Corderoy, R. (ed.): *Proceedings of the 15th Annual Conference of the Australasian Society for Computers in Learning in Tertiary Education*. Wollongong: 39-47.
- Albion, Peter (2000a). "Evaluating the implementation of problem-based learning in interactive multimedia". Paper presented at the *ASCILITE 2000 Conference*, Coffs Harbour.
- Albion, Peter (2000b): "Developing interactive multimedia using a problem-based learning framework". Paper presented at the *ASET/HERDSA Conference*, Toowoomba.
- Barab, Sasha A./Squire, Kurt D./Dueber, William (2000): "A co-evolutionary model for supporting the emergence of authenticity". *Educational Technology Research and Development*, 48 (2): 37-62.
- Barg, Mike/Crawford, Kathryn/Fekete, Alan/Greening, Tony/Hollands, Owen/Kay, Judy/Kingston, Jeffrey H. (2000): "Problem-Based Learning for Foundation Computer Science courses". *Computer Science Education* 10 (2): 1-20.
- Barrows, Howard S. (1986): "A taxonomy of problem-based learning methods". *Medical Education*, 20: 481-486.
- Boud, David (1985): "Problem-based learning in perspective." In: Boud, David (ed.): *Problem-Based Learning in Education for the Professions*. Sydney: 13-18.
- Broberg, Anders (2000): *Tools for Learners as Knowledge Workers* (PH.D. Thesis UMINF 00.01). Umeå University.
- Brown, John Seely/Collins, Alan/Duguid, Paul (1989): "Situated cognition and the culture of learning". *Educational Researcher* 18(1): 32-42.
- Brusilovsky, Peter (2001): "Adaptive Hypermedia". *User Modeling and User-Adapted Interaction* 11: 87-110.
- Camp, Gwendie (1996): "Problem-Based Learning: A Paradigm Shift or a Passing Fad?". *Medical Education Online* 1 (2): 2-6.
- Cavedon, Lawrence/Harland, James/Padgham, Lin (1997): "Problem Based Learning with Technological Support in an AI Subject: Description and Evaluation". In: Hurst, John (ed.): *Proceedings of the Second Australasian Conference on Computer Science Education, Melbourne*. New York: 191-200.
- Charlin, Bernard/Mann, Karen/Hansen, Penny (1998): "The many faces of problem-based learning: A framework for understanding and comparison". *Medical Teacher*, 20(4): 323-330.

- Clancey, William J. (1995): "A tutorial on situated learning". In: Self, J. (ed.): *Proceedings of the Internat. Conference on Computers and Education (Taiwan)*. Charlottesville, VA: 49-70.
- Dale, Robert/Mollá Aliod, Diego/Schwitler, Rolf (2003): "Natural Language Processing in the Undergraduate Curriculum". In: Greening, Tony/Lister, Raymond (eds.): *Proceedings of the Fifth Australasian Computing Education Conference (ACE2003)*. Adelaide, South Australia: 9-13.
- Farnsworth, C. (1994): "Using computer simulations in problem-based learning". In: Orey, M. (ed.): *Proc. of the Thirty-Fifth ADCIS Conference*. Nashville, TN: 137-140.
- Hoffman, Bob/Ritchie, Don (1997): "Using multimedia to overcome the problems with problem based learning". *Instructional Science*, 25(2): 97-115.
- Kinshuk/Patel, Ashok (1996): "Intelligent Tutoring Tools - Redesigning ITSs for adequate knowledge transfer emphasis". In: Lucas, C. (ed.): *Proc. of 1996 International Conference on Intelligent and Cognitive Systems*. Tehran: 221-226.
- Koch, Jürgen Hartmut/Teege, Gunnar (1999): "Problem Based Learning in Computer Science". In: Flückiger, F./Braun, T./Ninck, A. (eds.): *Proc. of 2nd International Conference on New Learning Technologies*. Bern, Switzerland.
see <http://citeseer.nj.nec.com/369966.html>
- Miao, Yongwu (2000): *Design and Implementation of a Collaborative Virtual Problem-Based Learning Environment*. PhD dissertation, Department of Computer Science, Darmstadt University of Technology.
- Norman, Geoffrey R./Schmidt, Henk G. (1992): "The psychological basis of problem-based learning: A review of the evidence". *Academic Medicine* 67(9): 557-565.
- Norman, Geoffrey R./Schmidt, Henk G. (2000): "Effectiveness of problem-based learning curricula: theory, practice and paper darts". *Medical Education* 34: 721-728.
- Patel, Ashok/Kinshuk (1997): "Intelligent tutoring tools on the internet - extending the scope of distance education". In: *18th International Conference on Distance Education (ICDE), June 2-6, 1997*. Pennsylvania, USA.
see <http://citeseer.nj.nec.com/patel97intelligent.html>
- Puntambekar, Sadhana (1999): "An integrated approach to individual and collaborative learning in a web-based learning environment". In: Hoadley, Christopher (ed.): *Proceedings of the CSCL (Computer Supported Collaborative Learning) Conference*. Palo Alto, Calif.: 160-171.
- Rhem, James (1998): "Problem-Based Learning: An Introduction". *The National Teaching & Learning Forum* 8 (1): 1-4.
- Savery, John R./Duffy, Thomas M. (1995): "Problem based learning: An instructional model and its constructivist framework". *Educational Technology* 35(5): 31-38.
- Seiler-Schiedt, Eva (2003): "E-Learning-Strategie - vier Jahre Erfahrung an der Universität Zürich". *SWITCHjournal* 1: 23-25.
- Weber, Gerhard/Specht, Marcus (1997): "User modeling and adaptive navigation support in WWW-based tutoring systems". *Proceedings of User Modeling '97*: 289-300.