Supporting Collaborative Web-Based Education via Annotations

Marián Šimko, Michal Barla, Vladimír Mihál, Maroš Unčík, Mária Bieliková
Institute of Informatics and Software Engineering,
Faculty of Informatics and Information Technologies,
Slovak University of Technology, Slovakia
{simko, barla}@fiit.stuba.sk, {xmihalv, xuncik}@is.stuba.sk, bielik@fiit.stuba.sk

Abstract: The spread of the internet and the evolution of the Web significantly change the way how we learn. Technology brings enhancements for learning almost on a daily basis. In this paper we discuss the role and importance of content annotations in a domain of web-based learning. We focus on collaboration support, continuous improvements of the content quality and increase of students' motivation. We present two basic types of activity flows within an adaptive web-based educational system and aim at annotations in both conceptual as well as practical point of view. We present results of several long-term experiments conducted in real courses of programming learning to observe the impact of annotations on educational process.

Introduction

Efficient education is a cornerstone of knowledge society. The key for further development of the knowledge society is technology-enhanced learning, allowing for anytime, anywhere access to learning materials. However, even if e-learning has many successful applications, the traditional in-class learning is still much superior way of education. The reason is in an implicit presence of two key elements: *adaptation* and *collaboration*, which are hard to implement in virtual learning environments. A good teacher adapts the explanations according to a feedback provided by the students (either to explicit one, when somebody raises his hand and asks a question, or implicit, recognizable from students' faces and reactions). A good teacher chooses the right (i.e., personalized) questions to determine how well a particular student masters a particular topic. However, the adaptation is almost always targeted to the whole class at once, usually considering the performance of majority of present students. This "one-size-fits-all" approach must inevitably fail for advanced students, who find the teacher's explanations slow and boring, as well as for weak students, who cannot keep pace with the majority of the class. Collaboration between students of a class can be also achieved very easily, when it is adequately supported by a teacher, e.g. by provoking a discussion between students themselves in order to make a common agreement about a particular problem (Alavi 1994).

The challenge is to incorporate these key elements of traditional in-class learning into web-based learning, which is currently the most popular way of technology-enhanced learning. Adaptive web-based learning occurs when an educational system adapts to user needs, goals and preferences (Beaumont & Brusilovsky 1995). Learning becomes more efficient and a learner is able to learn better or learn faster (Weibelzahl & Weber 2002). Hypertext-like nature of web-based education enables to overcome the mentioned "one-size-fits-all" problem of an in-class learning by tailoring the presented content to a particular learner. Personalization effect is often delivered in a form of recommendations or course sequencing that is based on techniques related to adaptive presentation and adaptive navigation support (Brusilovsky 1996). These techniques are based on a semantic descriptions of a subject domain (often forming a concept map) also referred to as a *course metadata*, and a *learner model* storing student's characteristics in relation to a domain model (course metadata).

We can already see efforts to incorporate aspects of *collaboration* into virtual learning environments. The emergence of web 2.0 principles such as user-oriented authoring (blogs, wikis), knowledge sharing and organization (annotating, tagging, discussing), and collaboration (instant messaging, social networks) reflect into learning as well. A web user (learner) is no longer considered to be a content consumer, but is given an opportunity to participate in the content creation and enhancement. Web 2.0 principles together with advancing web-based technologies improve overall user experience during learning by offering interaction, active participation and more competences. Besides greater autonomy for the learner, the traditional role of a teacher changes and distinction between teacher and student blurs (Downes 2005). The need for collaboration led to development of collaboration-supporting components or services for LMS (Meccawy et al. 2008) or attempts to extend existing learning standards (Ip & Canale 2003). Technology is leveraged in order to shift traditional individual learning towards collaborative learning (Dillenbourgh 1999, Tvarožek 2011).

Sustainability of the learning content quality, as one of the major bottlenecks of state-of-the-art educational web-based systems, is gaining popularity as we witness an incredible growth of learning materials available on the Web. The content may (and often does) contain errors that prevent from "smooth" learning. It is becoming heterogeneous and this reflects into various levels of appropriateness for a learner. For example, some parts are less and some parts are more difficult. In order to improve the quality of learning content, obvious question arises: How to develop and maintain the content, which can serve for better learning? How to handle big amounts of learning material? The situation is even worse when considering technology enhanced learning such as adaptive learning that relies on rich domain descriptions (metadata). Conceptual description of a domain contains – e.g. when considering concept map – hundreds of concepts (domain knowledge elements) and even thousands of relationships (Šimko & Bieliková 2009). Teachers are able to create a content, but they do not have enough time and space to define metadata. We believe these issues can be to some extent addressed by collaborative learning systems, where learners themselves can utilize a concept of *annotations* during learning. In this context we see annotations as a feasible tool for a domain model enrichment – not only by adding the content (user-created exercises, questions, etc.), but also by providing additional content descriptions (tags, comments, etc.). Furthermore, the potential of annotations exceeds the issue of content quality improvement; it also affects the motivation in learning. Enriched and interactive content can help to get and keep attention. We believe that visual and competitive attractiveness of annotations can increase learners' motivation resulting into improvement of learning performance.

In this paper we focus on annotation as a concept and tool for collaborative educational support. We point to its importance in the context of adaptive and collaborative web-based education. First we describe annotation as a whole. We discuss various types of annotations, their particular goals. We analyze annotations from user perspective: we cover user interfaces and usability. We emphasize the role of motivation within collaboration. We show how we integrated annotations into ALEF, adaptive learning framework aimed to improve learning efficiency and present results of several experiments conducted with selected annotation tools.

Collaborating by Annotating

Collaboration Flow in Learning

Collaborative learning can be understood in various ways. In this paper we adopt the broader definition, where collaborative learning is viewed as "a situation in which two or more people learn or attempt to learn something together" (Dillenbourg 1999). Learners work together to search for understanding, meaning or solutions to particular problems.

In our previous work we defined a concept of adaptive web-based learning 2.0 that merges adaptive learning with emerging concepts of Web 2.0 (Šimko et al. 2010). Considering such an environment, we distinguish two different groups of activities: *learning flow* and *collaboration flow* (Fig. 1). Although the flows are separated in the figure, in real world situations they occur simultaneously – a student can both learn and collaborate. The learning flow covers learning from study material delivered by a personalization engine – a *personalizer* module adapts the content (conceptually described by a *domain model*) to a particular user based on his actual knowledge, goals and needs (represented by a *user model*). Semantic logging and user model inferencing are aimed to collect and process data related to the usage of a system – from visiting a page to providing explicit feedback – and to update a user model.

Collaboration flow starts similarly – adapted content is presented to student, who can collaborate on improvements and enrichments of the content and metadata during learning sessions by using a number of *collaborative adaptive content creator* modules. Modules form the core of collaboration flow providing means for a collaboration support (creation, sharing or update of annotations).

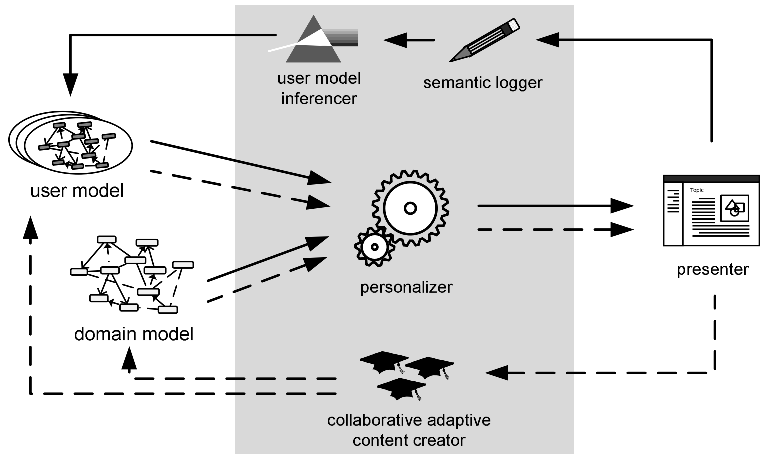

Figure 1: Activity flows during learning (Šimko et al. 2010).

We log student actions and derive semantics of those actions within each collaborative adaptive content creator module. The important aspect is evaluation of student behavior by analyzing logs in order to understand student motivation for actual creation of the annotation. Thus, the advantage of the course content annotations lies not only in the improvements of *domain model* quality, but also in more accurate estimation of user interests and current knowledge reflected in her *user model*.

Annotation framework

We implemented the collaborative flow within our Adaptive Learning Framework (ALEF). We put an emphasis on the design of annotations within the framework in order to allow for a straightforward creation of diverse types of annotations to enable and facilitate rich and multi-purpose participation and interaction.

ALEF's annotation framework was designed with respect to reusability and extendibility. In order to achieve it, we designed a common representation of content and annotations within the system. We see content and annotation as one entity – *Resource* (Fig. 2). Resources are connected with *Relationships* of various types determining their semantics. We support a general purpose relationship *Annotates*, that can be further specialized. Such a representation allows us to assign annotations to content, but even annotations to annotations. For example, this way we represent comments replies resulting into a discussion thread.

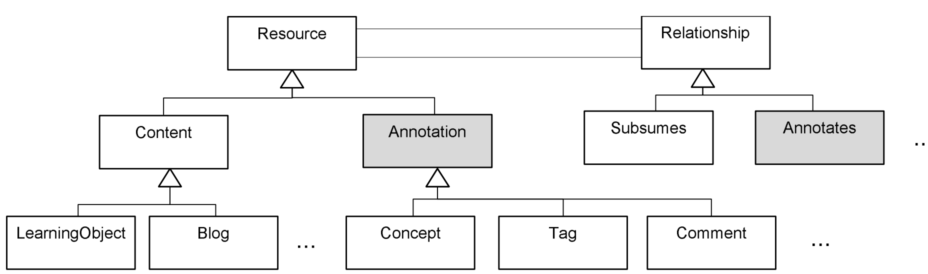

Figure 2: Extendibility of resource annotations. Resources can be assigned general purpose *Annotates* relationship that can be further specialized according to the specific needs.

Creating and Accessing Annotations – User Perspective

We recognize two different types of annotations from a conceptual point of view: *per-text-annotations* and *per-content-annotations.* The former is related to a specific part of text – e.g., a word, a phrase, a paragraph, which was selected during an annotation creation. The latter is related to a specific content as a whole – a question, an exercise, etc. Per-content annotations are typically created from the outside of the learning content using dedicated widgets.

Every annotation has two facets: *content* and *context*. Content is the actual payload of an annotation, inserted by a student, for instance a comment or URL of an external source. Context holds information related to association (binding) of an annotation to the learning object and optionally also to the text, where the annotation has been originally inserted by a student, if it is a *per-text-annotation*. Having two distinct facets, we provide students with two different types of navigation among annotations: *access-by-context* and *access-by-content*. The former is applicable only for *per-text-annotations* and is used to access an annotation within its context, i.e. in the text while reading it. The latter allows students to browse and view annotations separately from the text.

To create and access both content and context information of annotations, we designed four distinctive user interface elements:

- in-text interaction and presentation,
- sidebar,
- annotation browsers,
- annotation filter.

In-text interaction and presentation provides visualization and access of per-text annotations in their exact positions. To insert an annotation, a student selects part of a text and uses an in-text menu, which pops-up above the highlighted text (Fig. 3, left), chooses a type of annotation and inserts the actual content. Once an annotation is inserted, the selected text remains highlighted to indicate presence of an annotation, which can accessed by simply hovering the mouse over that text. The content of the annotation is shown in a popup window (Fig. 3, right). In-text interaction and presentation represents the fastest access to annotations with no significant interruption during learning.

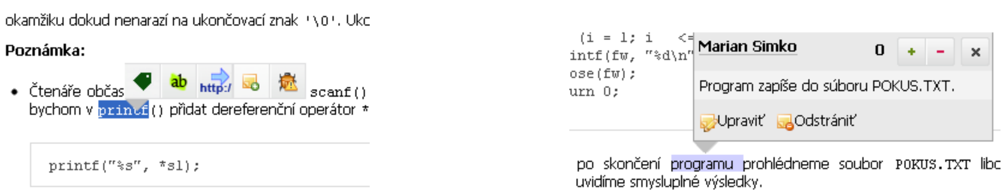

Figure 3: In-text menu for inserting new annotation and displaying an annotation (content in Slovak).

The drawback of in-text presentation of annotations is that the text become almost fully highlighted and considerably less readable as the amount of inserted annotations grows over time. We handle this situation by the annotation sidebar, which displays aggregations of regions with many annotations.

Sidebar provides an *access-by-context* navigation by visualizing annotated regions aside of positions of affected annotations within the text (Fig. 4). Regions on the sidebar are interactive; student can click on a region and view a list of all annotations within it. In-text style of visualization is used only when student interacts with certain annotation from the list, e.g. to view or edit its content.

Annotation browser provides the *access-by-content* navigation by listing all annotations related to the currently displayed learning object so students can easily read content of annotations regardless of their position within the text. Selection or interaction with an annotation inside the browser invokes in-text visualization to indicate context of an annotation, if any. Annotation browsers are implemented as widgets located on the right-side of the screen, not distracting from the main text in the central part.

Annotation filter allows users (students as well as teachers) to select which types of annotations they would like to have visible, e.g. to see only reported errors while fixing them. The filter of displayed annotations is a part of adaptability of the learning environment towards learners' preferences and actual needs.

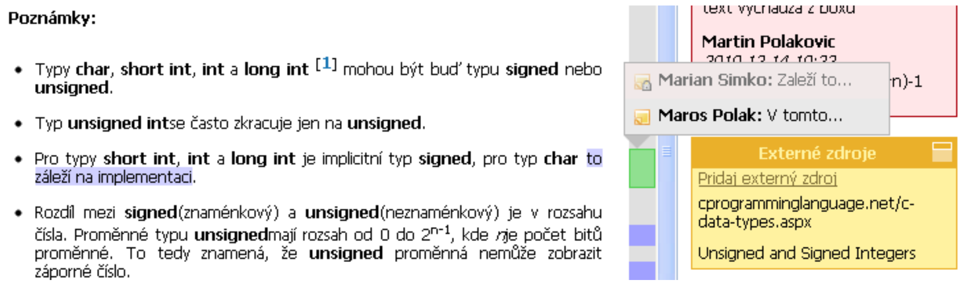

Figure 4: Annotation sidebar and annotation browsers (content in Slovak).

Collaborative Adaptive Content Creators

ALEF implements the annotation functionality within *collaborative adaptive content creator* components. We designed and implemented several such components (further referred to as *annotation widgets*). Each annotation widget introduces different goals for collaboration and may use any of aforementioned interface elements.

Tagger

Tagger is simple annotation widget that enables to assign user defined tags to the content. Motivation behind the tagging may differentiate among users. Some may use tags to categorize the content to their own categories, others to annotate content that they should pay more attention to in the future. Some may tag difficult learning objects, others learning objects that are important in order to achieve desired learning outcome. Tag annotations are realized as per-content annotations, i.e. a user can tag the whole learning object, not its particular parts. Users can assign private tags as well as public anonymous tags.

Besides providing additional style of navigation within a course, tags can be utilized for maintaining course metadata, as they represent a form of collaborative semantic descriptions. Moreover, user defined tags are very important for managing course quality. Tag analysis can reveal implicit user ratings of learning materials, thus allowing (later) filtering of popular, useful or well written learning objects.

Commentator

Commentator serves as a general purpose annotation widget for creation of per-text annotations. Students can add private, public or public but anonymous comments to any part of any learning object. Besides commenting the textual content, students can also comment ("reply to") other existing public or public anonymous comments, resulting into discussion threads on arbitrary topics, typically related to misconceptions or learning problems.

Comments as well as all of the following annotation types can be rated by other students. It helps us to distinguish more and less relevant contributions.

Error reporter

Error reporter constitutes generalization of commentator widget. It is specialized for inserting error and bug reports related to content, which are found by the students. Reported errors are evaluated by a teacher resulting into improved content serving for a better learning. This process supports collaboration between student and a teacher.

External resources inserter

External source inserter provides functionality for inserting links to external sources into learning content in order to enrich it with a quality source of knowledge. There are two ways to insert a link: either as a *per-text-annotation*, using in-text menu, or through external source widget. Context of an external source inserted through the widget will be assigned to whole learning text. After inserting external source, the widget will list external sources inserted into current learning object. Displayed sources are sorted by their quality, high quality sources are on top of the widget. Because of possibly large/considerable amount of inserted sources, we limited the number of visible sources to a default value. Student can eventually expand the widget to see all sources.

Questions creator

Question creator widget provides students with an interface for adding questions and for answering the questions added by their peers. This implicates that the system supports such types of question, which can be evaluated automatically. Currently, students may add five types of questions: (1) single choice question, (2) multiple choice question, (3) simple free text answer question, (4) sorting question (the task is to re-order the lines into correct order) and (5) text complement question (the task is to fill missing words into dedicated fields within the text, e.g. completing missing commands in a programming code). As a user-generated question is a special kind of an annotation, the procedure of adding a question is based on the principles of adding an annotation. The only difference is that after selecting the text and choosing a question as a type of annotation, a student is presented with a form for specifying question type, title, description and options along with an indication of correct answer(s).

The part dedicated for answering question created by students has similar interface as ordinary questions present within learning materials except that it is displayed within a widget instead of in a main content window when a student chooses a question. After filling an answer, it is automatically evaluated by the system and the student receives an instant feedback about her answer. Afterwards, she can rate the question in order to determine question's perceived quality. At the time, rating can be also viewed as a substitution for expressing other issues related to a question, e.g. addressing misconceptions.

Annotating, i.e. adding questions, is a complex process. It is necessary for given questions to exceed the minimal level of quality. We designed a question lifecycle to allow teacher to select it among the other most useful student-created questions and transform it into an expert (a teacher) questions to become a part of a curricula.

Rating Models for Motivation and Content Improvement

Motivation is an important factor affecting student's learning performance. In our educational system we motivate students by facing them with human basic instinct – competitiveness. We allow students to engage in a simple game of collecting *activity points* determined by type and intensity of actions that students perform within the system and to obtain highest rating among all peers. The hidden purpose of the game is to lead students to fully utilize collaboration and annotation options.

While students learn, they see their actual points and absolute position among other competing students. This is a sufficient information for a student to determine his chances to win or at least – to keep his current ranking.

Besides motivation, we introduce resource rating model. It partially influences student's rating in the system, but its major aim is to collect information about resource quality and usefulness that can be further utilized to improve the quality of learning content.

Student rating model

To increase motivation of students we designed a model of student rating. We introduce a *score* as an indicator of students' overall activity within the learning course. Our intent is to motivate students not to focus on single type of learning task, but to perform wide variety of tasks allowed by annotation widgets. To address this requirement, we separately evaluate score for each assessed learning activity. Score of an activity is computed as follows:

$$S(u,a) = k \cdot \ln\left(\left|\frac{S_B(a) \cdot N(u,a)}{k}\right| + 1\right)$$

where $S(u,a)$ is score for activity type a performed by a user u, $S_B(a)$ is a basic score for an activity type a, $N(u,a)$ is number of realizations of activity type a by user u, and k is a factor of distortion. We assess the following learning activities, categorized into two main flows (they may have different $S_B(a)$):

Learning flow
- reading (visiting) explanations
- answering questions
- solving exercises

Collaboration flow
- inserting annotations (comments, error reports, questions, external sources etc.)
- rating annotations

After evaluating scores for individual types of activities, we calculate a cumulative score reflecting overall activity of student. Besides particular learning and collaboration activities, student overall rating also depends on the quality of annotations he created:

$$S(u) = \sum_{a \in A} w_a \cdot S(u,a) + \sum_{r \in R_u} P_e(r)$$

where $S(u)$ is overall score rating of a student u, w_a denotes actual weight of an activity type a, $S(u,a)$ is score for activity type a performed by a student u, $P_e(r)$ is explicit rating of a resource r and R_u is a set of all resources created by user u. By using weights we set up different significance/usefulness of activity types for students.

Resource rating model

In order to determine a quality of a resource within a system (either content or annotation), we propose a resource quality rating model. Resource quality derives from the explicit resource rating given by students and implicit resource ranking derived from actions of students associated with a given resource. It is computed as weighted sum of both ratings:

$$P(r) = w_e \cdot P_e(r) + w_i \cdot P_i(r)$$

where $P(r)$ is overall rating of a resource r, $P_e(r)$ is explicit rating and $P_i(r)$ is implicit rating of a resource r, and w_e and w_i are weights for determining the importance of an explicit and implicit rating, respectively. $w_e + w_i = 1$. Explicit rating is common for all resource types and is computed as weighted mean of all users' ratings:

$$P_e(r) = \sum_{u \in U} K(u) \cdot P_e(u,r)$$

where $K(i)$ is overall knowledge level of a student u, $P_e(u,r)$ is explicit rating of resource r by a student u. By considering overall knowledge level of students, we prefer ratings given by more advanced students.

Computation of implicit rating depends on a type of resource. For each type of resource, different ratings can be defined according to actions related to that type of resource. The following table introduces main factors that influence the computation of implicit rating $P_i(r)$ for selected types of resources (Tab. 1). Note that implicit ranking has typically lower weight than explicit. Detailed description of each computation is out of the scope of this paper.

Table 1: Example factors influencing the computation of implicit rating $P_i(r)$.

Resource Type (*Content*)	**Example factors**	**Resource type (*Annotation*)**	**Example factors**
Explanation	Number of repeated visits	Tag	Popularity of a tag
Question	Ratio of correct to incorrect answers Number of *I don't understand* answers	User-created question	Ratio of correct to incorrect answers Number of *I don't understand* answers Number of errors (reported by error reporter)
Exercise	Number of hint requests	Comment	Number of replies
		Ext. resource	Number of accesses to external resource
		Error report	The relevance of error

Evaluation

We evaluated the proposed approach in a domain of programming learning. Due to the extensivity and complexity of the framework, we have not evaluated all aspects related to collaboration and learning content quality improvement based on annotations. However, we conducted three experiments in order to evaluate the following hypotheses:

1. Question creation supports/stimulates collaboration and course content enrichment.
2. Error reporter improves the quality of the learning content by revealing relevant errors in the content.
3. Motivating students based on a game increases learners activity in the system.

Question creation evaluation

We evaluated the first hypotheses in real educational settings during Functional and logic programming course lectured at Slovak University of Technology in Bratislava. Students had access to educational materials for learning programming language Prolog and were instructed to contribute questions to this learning content, and/or to answer and rate existing questions. We analyzed collected data (questions, answers, ratings) after seven days since the start of the experiment and found out that 30 students were involved in this activity (nearly 60% of all students that could participate). Students created together 88 questions and provided 660 answers.

We evaluated the hypothesis by assessing questions quality. We manually evaluated all obtained questions and divided them into three levels according to their quality (Tab. 2): First level contained high quality questions, which are about to be used as new educational materials. The second level contained correct questions but their difficulty wa rather low or they were not spelled properly. The third level contained questions useless for educational purposes, defective questions or ambiguous questions. The same sample of questions was evaluated by deriving ratings using question quality rating model based on learning object quality rating model. The method marked 24 questions of 88 as quality questions (Tab. 3). We compared these questions with the questions evaluated manually. According to the table the rating model did not estimate for quality questions high quality questions only, but also some questions with the second level of quality. Important result is that none of the faulty questions was marked as quality one.

Table 2: Manually determined quality level (88 questions).

Quality level	No. of questions
1. High quality questions	33
2. Medium quality questions	48
3. Defective questions	7

Table 3: Manually determined quality level*.

Quality level	No. of questions
1. High quality questions	17
2. Medium quality questions	7
3. Defective questions	0

* of 24 questions automatically marked as quality questions

The results show that even in a small period of a time (a week) student can leverage annotation-based tool to contribute new quality content. This way an overall learning content quality increases. More details on this experiment can be found in (Unčík & Bieliková 2010).

Error reporting evaluation

We evaluated error reporting feature during Procedural programming course lectured at Slovak University of Technology in Bratislava. For six weeks, students had possibility to create annotations for the learning content of the course (consisting of 213 explanations, 162 exercises and 420 questions). They created comments, bug reports and during the last two weeks of the semester they also got the possibility of inserting links to external resources. Out of 272 students, 129 created at least one annotation (47.42% of all students) .They added 949 annotations in total (Tab. 4), from which 782 were relevant (those that did not contain error or were semantically correct). Interesting finding is that the most common type of annotation was an error report. Students mostly revealed grammatical errors and misspellings that occurred in the learning content. Often duplication appeared among error reports, which increased the credibility of provided error reports.

An important aspect of error reporting is the level of knowledge of students who participate in annotating. We identified that 8% of students with the best overall assessment created 50% of all error reports and 20% of the most advanced students provided 82% error reports (similar statistics apply for all annotations). Such annotations also have higher quality and thereby help the weaker students. We also found that 5% of the most weak performing students did not create any annotation. Nevertheless, the number of provided relevant errors (either more or less severe) and average occurrence of errors (1 error per 1.46 learning objects) prove that error reporting feature reveals relevant errors and helps improving the quality of the learning content.

Table 4. Annotation statistics over six weeks of the term.

Annotation type	All	Relevant	Ratio
Comments	65	54	.83
Errors	697	546	.78
External sources*	187	182	.97
In total:	949	782	.82

* possible to add only during last two weeks of the term.

Motivation evaluation

In the last experiment we were observing how motivation influences students' behavior within a learning system. We used the same setup as in the previous case during six weeks (from 8th till 13th week of semester) in the course of learning programming. In the beginning of the course we encouraged students to use annotations in order to help themselves as well as others by improving the content. We also told the students that their activity and initiative will be in reward changed for bonus points to their overall assessment in the course. In the period half-time, at the beginning of eleventh week, we deployed the student rating model. Student activities are depicted in the following figure (Fig. 5).

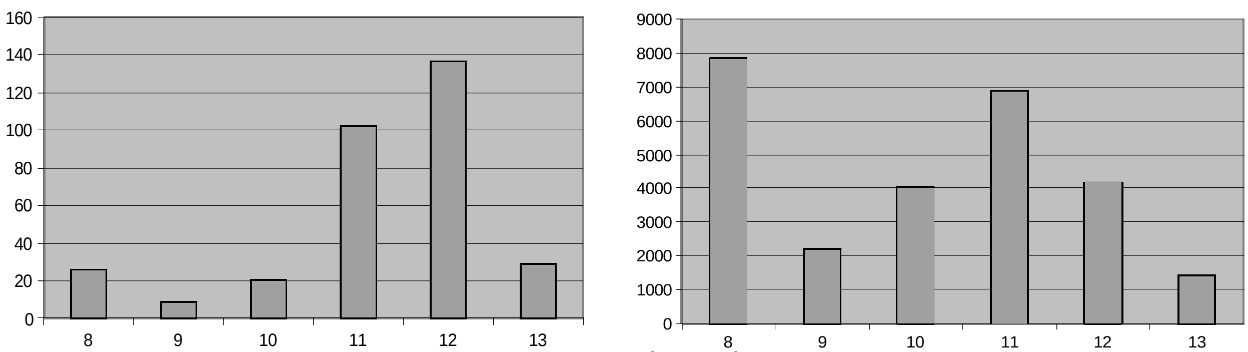

Figure 5: Student activity counts per day (y-axis) from 8th till 13th week of a term (x-axis). Activities related to annotations (left), activities related to content (right).

We observed increase in the count of specific learning activities after the deployment of student rating model. While there was no significant change in activities related to content (visiting an explanation, answering a question, etc.), the number of activities related to annotations (adding an annotation, rating an annotation) substantially increased.

We noticed decreased degree of activity in the 13th week, in both content and annotations related activities. This probably relates with the fact that it was the last week of the term and students were finishing their final projects, having not much time left for learning (in fact, at that time they were supposed to master majority of the topics). Further analysis showed that students' interest in more difficult and time-consuming content-related tasks (solving an exercise) increased more significantly and interest in simple and not demanding tasks was affected only slightly.

Considering the increase of activity, we believe that proposed score model increases motivation of students to solve variety of tasks, resulting into a potential increase of learning performance and outcomes.

Conclusions

User-generated annotations of the Web content is an active area of research, which already proved their positive impact in case of web-based learning. For instance, in (Farzan & Brusilovsky 2006), authors proposed a service for adding annotations to any set of linked web-documents, including learning materials and built an efficient social navigation support on top of it. However, similarly to other known works, they do not consider the annotation itself as a part of the content, meaning that learners cannot collaborate on annotations themselves, only on content initially generated by a teacher.

In this paper, we described our approach to integration of content annotations into adaptive web-based learning, where they play a crucial role of supporting the collaboration between students, leading to improvements of learning outcomes. Even more, collaboratively created annotations increase quality of available content either by enriching it by links to high quality external resources, pointing out key ideas in comments or just reporting a bug in the content.

Collaboration through annotations also positively influences students' motivation to actually learn through the system, as they feel to be involved in the process, playing an active role. We can additionally setup a gaming and reward mechanisms, which ensure continuous motivation to contribute quality annotations to the content.

We performed several long-term experiments in real courses of learning programming at bachelor degree of study. The results show that annotations can be used as a tool for supporting collaboration and learning content quality improvement. In order to evaluate the whole proposed concept of annotations and its binding to all layers of educational process, we need to (as a future work) conduct comprehensive multi-layer evaluation.

References

Alavi, Maryam. (1994). Computer-Mediate Collaborative Learning: An Empirical Evaluation. *MIS Quarterly*, 18(2), 159–174.

Beaumont, I. & Brusilovsky, P (1995). Adaptive educational hypermedia: From ideas to real systems. In H. Maurer (Ed.), *Proc. of ED-MEDIA '95 – World conf. on educational multimedia and hypermedia*. Graz, Austria, 93–98.

Brusilovsky, P. (1996) Methods and techniques of adaptive hypermedia. *User modeling and user-adapted interaction*. 6(2-3), 87–129.

Dillenbourg, P. (1999). Collaborative Learning: Cognitive and Computational Approaches. *Advances in Learning and Instruction Series*. New York, NY: Elsevier Science, Inc.

Downes, S. (2005). E-learning 2.0. *eLearn magazine*. Issue 10. ACM, p. 1.

Farzan R., & Brusilovsky P. (2006). AnnotatEd: A Social Navigation and Annotation Service for Web-based Educational Resources. *Proc. of E-Learn'06*, 2794–2802

Ip, A., & Canale, R. (2003). Supporting Collaborative Learning Activities with SCORM. *Educause in Australasia '03*, 669–678.

Meccawy, M., Blanchfield, P., Ashman, H., Brailsford, T., & Moore, A. (2008). Whurle 2.0: Adaptive Learning Meets Web 2.0. *Times of Convergence. Technologies Across Learning Contexts*, LNCS 5192, Springer. 274–279.

Šimko, M. Bieliková, M. (2009). Automated Educational Course Metadata Generation Based on Semantics Discovery. LNCS 5794, *Proc. of European Conf. on Technology Enhanced Learning*, EC TEL 2009, Nice, France. Springer, 99–105.

Šimko, M., Barla, M., & Bieliková, M. (2010). ALEF: A Framework for Adaptive Web-based Learning 2.0. N. Reynolds and M. Turcsányi-Szabó (Eds.): *KCKS 2010, IFIP Advances in Information and Communication Technology*, Vol. 324. Springer, pp. 367–378.

Tvarožek, J. (2011). Bootstrapping a Socially Intelligent Tutoring. *Information Sciences and Technologies Bulletin of the ACM Slovakia*. ISSN 1338-1237. 3(1), 33–41.

Unčík, M. & Bieliková, M. (2010). Annotating Educational Content by Questions Created by Learners. *Proc. of 5th Int. Workshop on Semantic Media Adaptation and Personalization*, SMAP 2010, IEEE, 13–18.

Weibelzahl, S., & Weber, G. (2002). Adapting to prior knowledge of learners. P. de Bra, P. Brusilovsky, R. Con-ejo (Eds.). *Proc.of the second int. conf. on Adaptive Hypermedia and Adaptive Web Based Systems, AH2002*, Springer, 448–451.

Acknowledgements

This work was partially supported by the grants VG1/0675/11, KEGA 028-025STU-4/2010 and it is the partial result of the Research & Development Operational Programme for the project Research of methods for acquisition, analysis and personalized conveying of information and knowledge, ITMS 26240220039, co-funded by the ERDF.